UNIVERSITY OF ZIMBABWE

COMPUTER SCIENCE

MSC 507 DISSERTATION

TITLE

Using wireless sensors for efficient data collection and optimum water usage in precision irrigation.

BY

MISHECK MARIMBI

Supervisors

Dr. Hapanyengwi (Computer Science, University of Zimbabwe)
Mr. M Munyaradzi (Computer Science, University of Zimbabwe)
Mr. B.Nyambo (Computer Science, University of Zimbabwe)

TABLE OF CONTENTS

ABSTRACT

The general ideology of a project is how to optimize water usage in irrigation based on the integration of measured soil moisture levels, temperature, and radiation, humidity and plant diameter. The research focused on the need for efficient data collection techniques from the wireless sensor nodes and analysis of this data for an efficient irrigation scheduling in order to optimize water usage. Issues which we looked into were the optimum placement of the sensor nodes, traffic aggregation and protocols for cooperative data forwarding. Models of different topologies were designed using different protocols and evaluated through simulations to come up with the best model that achieve optimum placement to minimize the number of nodes without compromising on the reading as well as improving data forwarding and aggregation. The model was cost effective as it displayed significant improvement in efficiency, power utilization and consumption and network lifetime and can be adopted for ordinary farmers in developing countries like Zimbabwe.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my Supervisors Dr. Hapanyengwi (Computer Science, University of Zimbabwe), Mr. M Munyaradzi (Computer Science, University of Zimbabwe), Mr. B.Nyambo (Computer Science, University of Zimbabwe) and Dr Mashonjowa (Physic, University of Zimbabwe) for their unwavering support throughout this research. Not forgetting Dr Nyandoro (Abilene University, Texas, USA) for his guidance at the start of this research, may the good Lord bless you all.

Above all I give glory to the Almighty God for seeing me through this whole research, blessed be His Holy name.

CHAPTER 1

1.0 INTRODUCTION

1.1Abstract

The general idea of this project is how to optimize water usage in irrigation based on the integration of measured soil moisture levels, temperature, and radiation, humidity and plant diameter. The research focused on the need for efficient data collection techniques from the wireless sensor nodes and analysis of this data for an efficient irrigation scheduling in order to optimize water usage. Issues which we looked into were the optimum placement of the sensor nodes, traffic aggregation and protocols for cooperative data forwarding. Models of different topologies were designed using different protocols and evaluated through simulations to come up with the best model that achieve optimum placement to minimize the number of nodes without compromising on the reading as well as improving data forwarding and aggregation. The model was cost effective as it displayed significant improvement in efficiency, power utilization and consumption and network lifetime and can be adopted for ordinary farmers in developing countries like Zimbabwe.

1.2Specific Research Questions/Sub-problems

1. How can we optimize water usage yet maximizing yield in Agriculture?
2. How can we achieve cost effective precision irrigation?
3. What are the contributing factors to effective precision irrigation?
4. How can we achieve efficient data collection using wireless sensor technology for efficient calculation of the relative evapo-transpiration?
5. Does placement of the wireless sensor nodes affect the efficiency of Data aggregation and forwarding and how can we achieve optimum placement for optimum data aggregation and forwarding?
6. Which protocols are most effective for our implementation?

1.3 Justification of the Study

1. Due to the scarcity and high cost of water in Zimbabwe there is definitely need for conservation of this precious resource, hence the need to optimize water usage and also improving yields. Many parts of Zimbabwe are dry and irrigation is the only reasonable

way of getting produce so this study can go along way into helping farmers achieve double benefit of water conservation and high yield.

2. Benefits

- **Safety**

  Wireless instruments can be used in locations that are difficult to access extreme conditions such as high temperature, pH, pressure, and so on. Using wireless sensors, operators can continuously supervise processes in hazardous environments and report the data back to an operator in a monitoring facility located at a safe distance away. Wireless measurement is also useful for obtaining data in hard to access locations.

- **Convenience**

  Wireless sensors can be used to form a web/network that would allow an engineer to monitor a number of different locations from one station. This provides a centralized control of a factory. Additionally, a number of wireless sensors have the ability to create a unique web page making up-to-the-minute data, accessible anywhere in the world.

- **Reduce Costs**

  Wireless process control can reduce the cost of monitoring and running a factory by eliminating the need for extension wire, conduit, and other costly accessories.

For efficient collection of data from the sensor nodes we need to have good traffic aggregation and protocols for cooperative data forwarding and also the optimum placement of the nodes to minimize the number of nodes yet maintaining data integrity and reducing cost. There is higher cost of infrastructure for wired network, like setting up cables and equipment through the whole field but we utilize the advent of modern digital wireless networks, hence the need for wireless sensors which are cost effective and more flexible in terms of placements.

Assumptions

I. Assume that simulation is a precise model of the real-time application

II. Assume that network will be efficient and reliable all the time.

Limitations of the Study

1. Wireless sensors were unavailable so we only ended up on simulation.

Feasibility of the Study

The study is feasible since even if actual implementation cannot be achieved, we can at least conduct the study under simulation. The UZ farm is readily available to conduct our experiments. We also have previous work in the field which can be incorporated in the study .We have experts in agricultural research who can help us with relevant information.

1.4 Problem statement

In many areas around the world, the success of agriculture is totally dependent on irrigation. Even in humid areas irrigation improves yields by supplying water to the crops in opportune situations. Water is a scarce resource that has to be managed wisely. Irrigation represents the major use of fresh water in nature and this need is increasing .Therefore there is need for more precision in irrigation technology.

Irrigation scheduling is a decision making process to determine the timing quantity of water applications, based on soil, crop and weather information. Although decades of research have been devoted to determining accurate scheduling methods the application of irrigation scheduling is still not wide spread. The acceptance of the technology b farmers is limited by the increase in quantity of actions and quality of decisions to be made on a daily basis. A solution to this problem is achievable through total automation.

An automated irrigation control system is a step beyond irrigation scheduling technology. Control systems require devices for sensing, communication data processing and actuating the system based on logical decision algorithm. Several different approaches for automatic irrigation control have been presented in recent years. Most of them rely on just one of the three basic components of the soil-plant-atmosphere system. (Reberto, 1998).

This study aims to develop an efficient wireless network Model for automatic irrigation scheduling techniques in irrigation systems using plant-soil-atmosphere wireless sensor nodes data as the indicator for determining the timing and amount of irrigation. The motive is to harness the wireless sensor technology advancements and advantages to the benefit of agriculture as a whole.

1.5 Research Purpose/Objective/Aims

The main objectives of this study will be to:

(i) Investigate the effect of optimum placement of the wireless sensor nodes to optimize data collection and accuracy.

(ii) To investigate on the most suitable protocols for cooperate data forwarding

(iii) To suggest efficient traffic aggregation techniques for effective data communication.

1.6 Hypothesis

Efficient estimation of the evapo-transpiration is dependent on the efficient collection of data needed to calculate the threshold from the sensor nodes.

1.7 Project description

I will start by analyzing the existing, protocols to come up with the best that suits our goal, the placement algorithms for optimum placement of the nodes, Different topologies for wireless sensor networks and come up with the best for our system. All these will be tested and analyzed under simulation and I will use the OMNeT 4.2.1simulator for all our analysis results .Different models of networks will be designed, simulated and the results tested under various stresses in order to have meaningful results. A reliable and efficient system will be designed using the most favorable protocols, placement algorithms, data aggregation techniques and then tested against the pre-existing ones.

1.8 Organization of the Study

Chapter 1

- Background to the study/the problem
- Objectives of the study
- Specific research questions
- Justification, the significance or importance of the study
- Hypotheses
- Assumptions guiding the study
- Scope of the study
- Delimitations of the study
- Limitations of the study
- Definition of terms

- Organization of the study

Chapter 2

Review of Related Literature

Chapter 3

Research Design and Methodology.

- Population
- The sample
- Sampling procedures
- Method(s) of data collection
- Data presentation, analysis and interpretation

Chapter 4

- ➢ Data presentation, analysis and interpretation
- ➢ Discussion of the Findings

Chapter 5

Summary, conclusions and recommendations

1.9 Expected results

A wireless sensor network powered system to collect and report real time irrigation data over large areas.

Desired Features:

- ➢ Low cost
- ➢ Scalable
- ➢ Reliable
- ➢ Easily deployed
- ➢ Low power and low maintenance

1.10 Resources

Available Resources

- ➢ Computer for data capturing and server
- ➢ OMNeT ++ simulation software

CHAPTER 2

2.1 LITERATURE REVIEW

Irrigation overview in Zimbabwe

Zimbabwe's agricultural sector is the second largest forces driving the economy being the second largest foreign currency earner. Zimbabwe is struggling to retain its status as a bread basket of Southern Africa. Climate changes introduced greater variability in crop yields, thus making crop production a more risky agricultural activity,(FC Mwamuka, 1999).Irrigation will boost crop production .Climate change is slowly taking place .This change will result in impacts whose direction, timing and path are neither understood nor accurately predictable. In recent years, drought has strained farmers and pastoralists, and the land reform and resettlement program has created an increased need for the development of irrigation systems for smallholders. Zimbabwe has well-developed dams, but they have not been fully exploited. Beginning in the 1990's the government recognized the need for a new framework governing water resources, and the importance of providing irrigation for smallholders in order to increase agricultural productivity. (Makwiro et al, 1999)There is thus need for sustained scientific researches to enable the prediction of climate change hence we can begin to develop and implement the most appropriate resource management strategies and technologies to combat the impacts of climate change on the agricultural sector (J.M Makacho, 1999). In 2002, total water withdrawal in Zimbabwe was approximately 4.2 cubic kilometers. Seventy-nine percent of this water was used for agriculture, including irrigation, fish-farming and livestock. Zimbabwe has an estimated 550,000 hectares that are irrigable, but irrigation systems have been developed for only 200,000 hectares. Of the developed systems, many have deteriorated or been destroyed in the years of conflict related to land reform efforts, (Makwiro et al, 1999) 90 % use traditional irrigation schemes and furrow irrigation.

Zimbabwe has total annual internal renewable water resources of 12.26 cubic kilometers: 11.26 cubic kilometers are surface water resources and 6 cubic kilometers are groundwater, with an estimated 5 cubic kilometer overlap between the two sources. The country has an average annual rainfall of 657 millimeters, but rainfall can range from over 1000 to only 300 to 450 millimeters, depending on location. Rainfall figures decrease steadily across the country from north to south and also from east to west. Thirty-seven percent of the country can sustain rain-fed agriculture, while the remainder is dependent on supplemental or full-time irrigation. (Makwiro et al, 1999)

There is need to resuscitate the irrigation schemes and apply new technology especially those which specialize in precision irrigation.

2.2.0 Factors influencing Irrigation

There many factors which contribute to plant growth, some contribute directly but others indirectly. We will only focus on the major factors because exhaustion of all of them is impractical. The key fact we want to determine here, which will help us calculate irrigation time and rate is the amount of water of water lost by the plant. In precision irrigation the objective is to replace only the amount of water that the plant has lost without over irrigating or under irrigating thus precision. We can even incorporate factors from the weather station like annual rainfall, wind speed and so on; different formulas have been used to calculate evapotranspiration taking different factors into consideration. Evapotranspiration is usually expressed in millimeters per unit of time, for example. Mm/day, mm/month, or mm/season. The accuracy of this figure is much important because if it’s not it can lead to over irrigation or under irrigation which causes stress on the plants. The evapotranspiration can be determined experimentally using an evaporation pan or theoretically using measured climatic data.

The crop water need is defined as the depth amount of water needed to meet the water loss through evapotranspiration. In other words, it is the amount of water needed by the various crops to grow optimally. The crop water need always refers to a crop grown under optimal conditions that is a uniform crop, actively growing, completely shading the ground, free of diseases, and favorable soil conditions (including fertility and water). The crop thus reaches its full production potential under the given environment.

The crop water need mainly depends on:

- the climate: in a sunny and hot climate crops need more water per day than in a cloudy and cool climate
- the crop type: crops like maize or sugarcane need more water than crops like millet or sorghum

- The growth stage of the crop; fully grown crops need more water than crops that have just been planted.

Soil moisture

Soil moisture depends on the precipitation, water retention capacity, temperature, surface evaporation so it's a good measure to determine the rate of evaporation and the availability of the water to the plant roots. If soil moisture decreases faster it means the plant is also losing water at faster rate. We would take the rate of decrease of soil moisture is proportional to the rate of water lost by the plant.

Air humidity

While the energy supply from the sun and surrounding air is the main driving force for the vaporization of water, the difference between the water vapor pressure at the evapotranspiring surface and the surrounding air is the determining factor for the vapor removal. Well-watered fields in hot dry arid regions consume large amounts of water due to the abundance of energy and the desiccating power of the atmosphere.

Wind speed

The process of vapor removal depends to a large extent on wind and air turbulence which transfers large quantities of air over the evaporating surface. When vaporizing water, the air above the evaporating surface becomes gradually saturated with water vapor. If this air is not continuously replaced with drier air, the driving force for water vapor removal and the evapotranspiration rate decreases.

Air temperature

The solar radiation absorbed by the atmosphere and the heat emitted by the earth increase the air temperature. The sensible heat of the surrounding air transfers energy to the crop and exerts as such a controlling influence on the rate of evapotranspiration. In sunny, warm weather the loss of water by evapotranspiration is greater than in cloudy and cool weather.

2.3 Empirical formulae and methods

The Penman Equation

In 1948, Howard Penman combined the energy balance with the mass transfer method and derived an equation to compute the evaporation from an open water surface from standard climatologically records of sunshine, temperature, humidity and wind speed.

Penman (1948) defined Ea empirically as

$$E_a = W_f \ (e^o - e_a) \qquad (1)$$

Where Ea is in mm d^{-1}

W_f is called a wind function in mm d-1 kPa-1 [typically expressed as a linear function of wind speed in m s-1 (Uz) at the reference height (z) above the ground]

e^o is the saturated vapor pressure in kPa at mean air temperature, and e^a is mean ambient vapor pressure in kPa at the reference height above ground [ea = RH eo, where RH is mean relative humidity as a fraction; conceptually, ea should equal the saturated vapor pressure at the daily mean dew point temperature].

[e^a = RH e^o, where RH is mean relative humidity as a fraction; conceptually, e^a should equal the saturated vapor pressure at the daily mean dew point temperature]

Penman noted in his 1948 paper one of the experimental problems needing a solution was the reliable estimation of the daily mean dew point temperature. This problem has led to current differences in using Penman's equation and has resulted in myriad different versions of a "modified

Penman equation" with varying wind functions and methods for estimating mean daily vapor pressure deficit $(e^0 - e^a)$(A. Elusoji et al, 2011).

It is critical to build the Penman-Monteith equation first on an understanding of the Penman equation and its subtleties. Penman (1948) defined E as open water evaporation. He expressed bare-, wet-soil evaporation or grass evaporation, E_o, (we now call this evapotranspiration, especially in the U.S.) as fractions of open water evaporation (Ew)

[That is, Eo = f Ew, where f is expressed as a fraction].

The "f" values he measured typically varied from about 0.5-0.6 in winter to near 0.8-1.0 in summer. Grass evaporation "f" values were slightly larger than "f" values for bare soil with a water table near the surface (120 to 400 mm beneath the soil surface).

The Penman equation, therefore, only required routine weather observations (although some measurements like wind speed and cloud cover were not available everywhere) from a single level or height above ground. But the theory was rather advanced for its time. Without computers to perform the tedious computations, most engineers continued to rely on simpler evapotranspiration (ET) estimation methods such as the Blaney Criddle, Thornthwaite, or Jensen. One of the earliest uses of the Penman equation in the U.S. was by Van Bavel (1956) for irrigation scheduling. Another advance to aid the use of the Penman equation was a wider acceptance and familiarity with metric units or the S.I. unit system that greatly streamlined the cumbersome original English units used in 1948.

The Penman-Monteith Equation

This so-called combination method was further developed by many researchers and extended to cropped surfaces by introducing resistance factors. The Penman-Monteith method refers to the use of an equation for computing water evaporation from vegetated surfaces. It was proposed and developed by John Monteith in his seminal paper (Monteith, 1965) in which he illustrated its thermodynamic basis with a psychometric chart (a graph of vapor pressure at various relative saturations versus air temperature at a known air pressure). Monteith's derivation was built upon that of Howard Penman (Penman, 1948) in the now well-known combination equation (so named based on its "combination" of an energy balance and an aerodynamic formula) given as

$$\lambda ET = \frac{\Delta(R_n - G) + \rho_a c_p \frac{(e_s - e_a)}{r_a}}{\Delta + \gamma\left(1 + \frac{r_s}{r_a}\right)} \quad (2)$$

Where R_n is the net radiation,

G is the soil heat flux, (e_s - e_a) represents the vapor pressure deficit of the air,

r_a is the mean air density at constant pressure,

c_p is the specific heat of the air,

D represents the slope of the saturation vapor pressure temperature relationship,

g is the psychometric constant,

And r_s and r_a are the (bulk) surface and aerodynamic resistances

The **Penman-Monteith** equation requires daily mean temperature, wind speed, relative humidity, and solar radiation to predict net evapotranspiration. Other than radiation, these parameters are implicit in the derivation of Δ, c_p, and δ_q, if not conductance below.

g_a is Conductivity of air, atmospheric conductance (m s^{-1}) and $g_a = 1/ r_a$

g_s = Conductivity of stoma, surface conductance (m s^{-1}) and $g_s = 1/ r_s$

The Penman-Monteith approach as formulated above includes all parameters that govern energy exchange and corresponding latent heat flux (evapotranspiration) from uniform expanses of vegetation. Most of the parameters are measured or can be readily calculated from weather data. The equation can be utilized for the direct calculation of any crop evapotranspiration as the surface and aerodynamic resistances are crop specific.

Solar radiation

The evapotranspiration process is determined by the amount of energy available to vaporize water. Solar radiation is the largest energy source and is able to change large quantities of liquid water into water vapor. The potential amount of radiation that can reach the evaporating surface is determined by its location and time of the year. Due to differences in the position of the sun, the potential radiation differs at various latitudes and in different seasons. The actual solar radiation reaching the evaporating surface depends on the turbidity of the atmosphere and the presence of clouds which reflect and absorb major parts of the radiation. When assessing the effect of solar radiation on evapotranspiration, one should also bear in mind that not all available energy is used to vaporize water. Part of the solar energy is used to heat up the atmosphere and the soil profile

Extraterrestrial radiation (R_a)

The radiation striking a surface perpendicular to the sun's rays at the top of the earth's atmosphere, called the solar constant, is about 0.082 MJ m^{-2} min^{-1}. The local intensity of radiation is, however, determined by the angle between the direction of the sun's rays and the normal to the surface of the atmosphere. This angle will change during the day and will be different at different latitudes and in different seasons. The solar radiation received at the top of the earth's atmosphere on a horizontal surface is called the extraterrestrial (solar) radiation, R_a.

If the sun is directly overhead, the angle of incidence is zero and the extraterrestrial radiation is 0.0820 MJ m^{-2} min^{-1}. As seasons change, the position of the sun, the length of the day and, hence, R_a change as well. Extraterrestrial radiation is thus a function of latitude, date and time of day.

Solar or shortwave radiation (R_s)

As the radiation penetrates the atmosphere, some of the radiation is scattered, reflected or absorbed by the atmospheric gases, clouds and dust. The amount of radiation reaching a horizontal plane is known as the solar radiation, R_s. Because the sun emits energy by means of electromagnetic waves characterized by short wavelengths, solar radiation is also referred to as shortwave radiation.

For a cloudless day, R_s is roughly 75% of extraterrestrial radiation. On a cloudy day, the radiation is scattered in the atmosphere, but even with extremely dense cloud cover, about 25% of the extraterrestrial radiation may still reach the earth's surface mainly as diffuse sky radiation. Solar radiation is also known as global radiation, meaning that it is the sum of direct shortwave radiation from the sun and diffuse sky radiation from all upward angles.

Relative shortwave radiation (R_s/R_{so})

The relative shortwave radiation is the ratio of the solar radiation (R_s) to the clear-sky solar radiation (R_{so}). R_s is the solar radiation that actually reaches the earth's surface in a given period, while R_{so} is the solar radiation that would reach the same surface during the same period but under cloudless conditions.

The relative shortwave radiation is a way to express the cloudiness of the atmosphere; the cloudier the sky the smaller the ratio. The ratio varies between about 0.33 (dense cloud cover) and 1 (clear sky). In the absence of a direct measurement of R_n, the relative shortwave radiation is used in the computation of the net long wave radiation.

Relative sunshine duration (n/N)

The relative sunshine duration is another ratio that expresses the cloudiness of the atmosphere. It is the ratio of the actual duration of sunshine, n, to the maximum possible duration of sunshine or daylight hours N. In the absence of any clouds, the actual duration of sunshine is equal to the daylight hours (n = N) and the ratio is one, while on cloudy days n and consequently the ratio may be zero. In the absence of a direct measurement of R_s, the relative sunshine duration, n/N, is often used to derive solar radiation from extraterrestrial radiation.

As with extraterrestrial radiation, the day length N depends on the position of the sun and is hence a function of latitude and date.

Albedo (A_b) and net solar radiation (R_{ns})

A considerable amount of solar radiation reaching the earth's surface is reflected. The fraction, a, of the solar radiation reflected by the surface is known as the albedo. The albedo is highly variable for different surfaces and for the angle of incidence or slope of the ground surface. It may be as large as 0.95 for freshly fallen snow and as small as 0.05 for a wet bare soil. A green vegetation cover has an albedo of about 0.20-0.25. For the green grass reference crop, $\mathbf{A_b}$ is assumed to have a value of 0.23.

The net solar radiation, R_{ns}, is the fraction of the solar radiation R_s that is not reflected from the surface. Its value is $(1 - A_b)R_s$.

Net long wave radiation (R_{nl})

The solar radiation absorbed by the earth is converted to heat energy. By several processes, including emission of radiation, the earth loses this energy. The earth, which is at a much lower temperature than the sun, emits radioactive energy with wavelengths longer than those from the sun. Therefore, the terrestrial radiation is referred to as long wave radiation. The emitted long wave radiation ($R_{l, up}$) is absorbed by the atmosphere or is lost into space. The long wave radiation received by the atmosphere ($R_{l, down}$) increases its temperature and, as a consequence, the atmosphere radiates energy of its own. Part of the radiation finds its way back to the earth's surface. Consequently, the earth's surface both emits and receives long wave radiation. The difference between outgoing and incoming long wave radiation is called the net long wave radiation, R_{nl}. As the outgoing long wave radiation is almost always greater than me incoming long wave radiation, R_{nl} represents an energy loss.

Net radiation (R_n)

The net radiation, R_n, is the difference between incoming and outgoing radiation of both short and long wavelengths. It is the balance between the energy absorbed, reflected and emitted by the earth's surface or the difference between the incoming net shortwave (R_{ns}) and the net outgoing long wave (R_{nl}) radiation. R_n is normally positive during the daytime and negative during the nighttime. The total daily value for R_n is almost always positive over a period of 24 hours, except in extreme conditions at high latitudes.

Soil heat flux (G)

In making estimates of evapotranspiration, all terms of the energy balance should be considered. The soil heat flux, G, is the energy that is utilized in heating the soil. G is positive when the soil is warming and negative when the soil is cooling. Although the soil heat flux is small compared to R_n and may often be ignored, the amount of energy gained or lost by the soil in this process should theoretically be subtracted or added to R_n when estimating evapotranspiration.

'Bulk' surface resistance (r_s)

The 'bulk' surface resistance describes the resistance of vapor flow through the transpiring crop and evaporating soil surface. Where the vegetation does not completely cover the soil, the resistance factor should indeed include the effects of the evaporation from the soil surface. If the crop is not transpiring at a potential rate, the resistance depends also on the water status of the vegetation

Air temperature

The solar radiation absorbed by the atmosphere and the heat emitted by the earth increase the air temperature. The sensible heat of the surrounding air transfers energy to the crop and exerts as such a controlling influence on the rate of evapotranspiration. In sunny, warm weather the loss of water by evapotranspiration is greater than in cloudy and cool weather.

The FAO Penman-Monteith Equation

By defining the reference crop as a hypothetical crop with an assumed height of 0.12 m having a surface resistance of 70 s m^{-1} and an albedo of 0.23, closely resembling the evaporation of an extension surface of green grass of uniform height, actively growing and adequately watered, the FAO Penman-Monteith method was developed. The method overcomes shortcomings of the previous FAO Penman method and provides values more consistent with actual crop water use data worldwide.

From the original Penman-Monteith equation and the equations of the aerodynamic and surface resistance, the FAO Penman-Monteith method to estimate ET_o can be derived.

$$ET_o = \frac{0.408\Delta(R_n - G) + \gamma \frac{900}{T+273} u_2 (e_s - e_a)}{\Delta + \gamma(1 + 0.34 u_2)} \qquad (3)$$

Where

ET_o reference evapotranspiration [mm day^{-1}],
R_n net radiation at the crop surface [MJ m^{-2} day^{-1}],
G soil heat flux density [MJ m^{-2} day^{-1}],
T mean daily air temperature at 2 m height [°C],
u_2 wind speed at 2 m height [m s^{-1}],
e_s saturation vapor pressure [kPa],
e_a actual vapor pressure [kPa],
e_s - e_a saturation vapor pressure deficit [kPa],
□ slope vapor pressure curve [kPa $°C^{-1}$],
□ psychometric constant [kPa $°C^{-1}$].

Blaney-Criddle Method

If no measured data on pan evaporation are available locally, a theoretical method (for example. the Blaney-Criddle method) to calculate the reference crop evapotranspiration ETo has to be used. There are a large number of theoretical methods to determine the ETo. Many of them have been determined and tested locally. If such local formulae are available they should be used. If such local formulae are not available one of the general theoretical methods has to be used. The most commonly used theoretical method is the modified Penman method which is described in detail in FAO Irrigation and Drainage Paper 24. This method, however, is rather complicated and beyond the scope of this manual.

Here only the Blaney-Criddle method is given. The Blaney-Criddle method is simple, using measured data on temperature only. It should be noted, however, that this method is not very accurate; it provides a rough estimate or "order of magnitude" only. Especially under "extreme" climatic conditions the Blaney-Criddle method is inaccurate: in windy, dry, sunny areas, the ETo is underestimated (up to some 60 percent), while in calm, humid, clouded areas, the ETo is overestimated (up to some 40 percent).

The Blaney-Criddle method

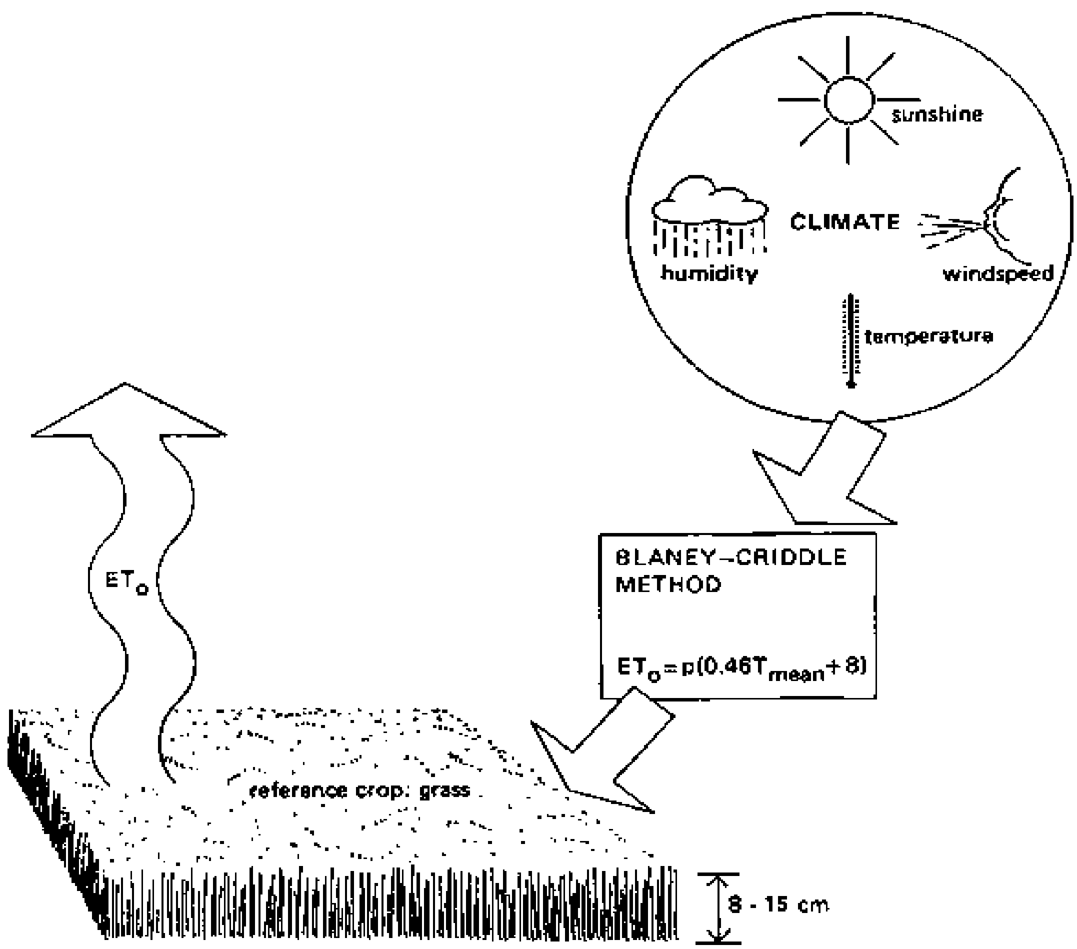


Fig. 2.0 Blaney-Criddle's Model

The Blaney-Criddle formula: ETo = p (0.46 T mean +8)

ETo = Reference crop evapotranspiration (mm/day) as an average for a period of 1 month
T mean = mean daily temperature (°C)
p = mean daily percentage of annual daytime hours

The use of the Blaney-Criddle formula

Step 1**: Determination of the mean daily temperature: T mean**

The Blaney-Criddle method always refers to mean **monthly** values, both for the temperature and the ETo. If, for example, it is found that T mean in March is 28°C, it means that during the whole month of March the mean daily temperature is 28°C.

If in a local meteorological station the daily minimum and maximum temperatures are measured, the mean daily temperature is calculated as follows:

$$\text{T max} = \frac{\text{sum of all T max values during the month}}{\text{number of days of the month}}$$

$$\text{T min} = \frac{\text{sum of all T min values during the month}}{\text{number of days of the month}}$$

$$\text{T mean} = \frac{\text{T max} + \text{T min}}{2}$$

Step 2: **Determination of the mean daily percentage of annual daytime hours: p**

To determine the value of p. Table 4 is used. To be able to determine the p value it is essential to know the approximate latitude of the area: the number of degrees north or south of the equator suppose the p value for the month **March** has to be determined for an area with latitude of 45° south. From Table 4 it can be seen that the p value during March = 0.28.

Step 3**: Calculate ETo,** using the formula: ETo = p (0.46 T mean + 8)

For example, when p = 0.29 and T mean = 21.5°C the ETo is calculated as follows:

ETo = 0.29 (0.46 × 21.5 + 8) = 0.29 (9.89 + 8) = 0.29 × 17.89 = 5.2 mm/day. (Makwiro et al, 1999)

2.4 History of Wireless Sensor Technology

The history of sensor networks spans four phases.

Phase 1: Cold-War Era Military Sensor Networks .During the cold war, extensive acoustic networks were developed in the United States for submarine surveillance; some of these sensors are still being used by the National Oceanographic and Atmospheric Administration (NOAA) to monitor seismic activity in the ocean. Also, networks of air defense radars were deployed to cover North America; to handle this, a battery of Airborne Warning and Control System (AWACS) planes operated as sensors.

Phase 2: Defense Advanced Research Projects Agency Initiatives .The major impetus to research on sensor networks took place in the early 1980s with programs sponsored by the Defense Advanced Research Projects Agency (DARPA). The distributed sensor networks (DSN) work aimed at determining if newly developed TCP–IP protocols and ARPAnet's (the predecessor of the Internet) approach to communication could be used in the context of sensor networks. DSN postulated the existence of many low-cost spatially distributed sensing nodes that were designed to operate in a collaborative manner, yet be autonomous; the goal was for the network to route information to the node that can best utilize the information. The DSN program focused on distributed computing, signal processing, and tracking. Technology elements included acoustic sensors, high-level communication protocols, processing and algorithm calculations (for example., self-location algorithms for sensors), and distributed software (dynamically modifiable distributed systems and language design). Researchers at Carnegie Mellon University focused on providing a network operating system for flexible transparent access to distributed resources, and researchers at the Massachusetts Institute of Technology focused on knowledge-based signal-processing techniques. Test beds were developed for tracking multiple targets in a distributed environment; all components in the test bed network were custom built. Ongoing work in the 1980s resulted in the development of a multiple-hypothesis tracking algorithm to address difficult problems involving high target density, missing detections, and false alarms; multiple-hypothesis tracking is now a standard approach to challenging tracking problems.

Phase 3: Military Applications Developed or Deployed in the 1980s and 1990s (These can properly be called first-generation commercial products.) Based on the results generated by the DARPA–DSN research and the test beds developed, military planners set out in the 1980s and 1990s to adopt sensor network technology, making it a key component of network-centric warfare.

Phase 4: Present-Day Sensor Network Research (These can properly be called second-generation commercial products.) Advances in computing and communication that have taken place in the late 1990s and early 2000s have resulted in a new generation of sensor network technology. Evolving sensor networks represent a significant improvement over traditional sensors. Inexpensive compact sensors based on a number of high-density technologies, including MEMS and (in the next few years) nano-scale electromechanical systems (NEMS), are appearing. Standardization is a key to wide-scale deployment of any technology, including WSN (for example., Internet–Web, MPEG-4 digital video, wireless cellular, VoIP). Advances in IEEE 802.11a/b/g-based wireless networking and other wireless systems such as Bluetooth, ZigBee, 9 and WiMax are now facilitating reliable and ubiquitous connectivity. Inexpensive processors that have low power-consumption requirements make possible the deployment of sensors for a plethora of applications. Commercially-focused efforts are now directed at defining mesh, peer-to-peer, and cluster-tree network topologies with data security features and interoperable application profiles. (Sohraby et al ,2007)

2.5 What are Wireless Sensors?

Wireless sensors are standard measurement tools equipped with transmitters to convert signals from process control instruments into a radio transmission. The radio signal is interpreted by a receiver which then converts the wireless signal to a specific, desired output, such as an analog current or data analysis via computer software. WSN is usually developed for a particular application at low power and low cost. It usually does not require a complex, general-purpose operating system such as Microsoft Windows or Linux. The operating system for WSN is similar to an embedded system. TinyOS is an operating system specifically designed for WSN based on an event driven programming model. TinyOS supports nesC programming language which is built as an extension to the C programming language. LiteOS is another operating system developed for WSN and it supports C programming as well. Contiki is an operating system for WSN, which uses a simpler programming style in C (Sohraby et al, 2007). All sensors are reconfigurable as per the stages of crop growth, dynamic changes in the targeted area, nature of soil, climate, season and type of crop are taken into consideration. (Wei Han, 2011)

2.6 Sensor Node Architecture

Due to their diversity there is no standard node architecture but a typical sensor node comprise of 5 main components

1. Controller –to process the relevant data ,capable of executing arbitrary code.(for example. Atmel ATmega 128 L)
2. Memory-to store programs and intermediate data for the program and data that is. RAM,EPROM,EEPROM,Flash and so on
3. Sensors and actuators- these are devices that can observe or control physical parameters of the environment .they are actual interface to the physical world.
4. Communication Devices-for sending and receiving information over a wireless channel.eg transceivers
5. Power Supply- to supply power to the electrical components

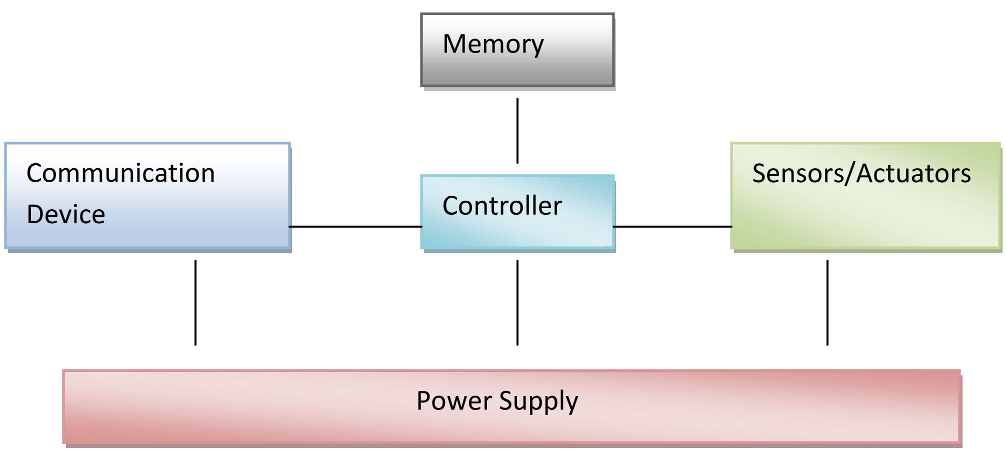


Fig 2.1 Node Architecture

Wireless Sensor Networks(WSN) emerged as an alternative class of networks which counteract the high cost of wiring ,maintenance problem, and emobility.WSN consists of individual nodes that can interact with their environment by sensing or controlling physical parameters .These nodes have to collaborate to fulfill their task ,a single node is capable of doing so and they use wireless communication to enable the collaboration.WSN are powerful in that they are amenable to support a lot of different applications.

Unlike cell phone systems that deny service when too many phones are active in a small area, the interconnection of a wireless sensor network only grows stronger as nodes are added. As long as there is sufficient density, a single network of nodes can grow to cover limitless area. Each node has a communication range of 50 meters.

2.7 Wireless Sensor Network (WSN) overview

WSN can operate in a wide range of environments and provide advantages in cost, size, power, flexibility and distributed intelligence compared to wired ones. A WSN is a system comprised of radio frequency (RF) transceivers, sensors, microcontrollers and power sources. Recent advances in wireless sensor networking technology have led to the development of low cost, low power, multifunctional sensor nodes. Sensor nodes enable environment sensing together with data processing. Instrumented with a variety of sensors, such as temperature, humidity and volatile compound detection, allow monitoring of different environments. They are able to network with other sensor systems and exchange data with external users. A general WSN protocol consists of Application layer, transport layer, physical layer, power management plane, mobility management plane and task management plane. (Ruiz et al 2009).

WSN allows different network topologies and multi hop communication. Each wireless sensor node communicates with a gateway unit which can communicate with other computers via other networks such as LAN, WLAN, CAN, WWAN, Internet, using protocols like GSM and GPRS.

A wireless sensor network (WSN) consists of spatially distributed autonomous sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants. The development of wireless sensor networks was motivated by military applications such as battlefield surveillance. They are now used in many industrial and civilian application areas, including industrial process monitoring and control, machine health monitoring, environment and habitat monitoring, healthcare applications, home automation, and traffic control.

In addition to one or more sensors, each node in a sensor network is typically equipped with a radio transceiver or other wireless communications device, a small microcontroller, and an energy source, usually a battery. Sensor networks are the key to gathering the information needed by smart environments, whether in buildings, utilities, industrial, home, shipboard, transportation systems automation, or elsewhere.

2.7.1 Fields of application of wireless sensor networks

There are numerous different fields of application of sensor networks. For example, forest fires can be detected by sensor networks so that they can be fought at an early stage. Sensor networks can be used to monitor the structural integrity of civil structures by localizing damage for example in bridges. Further, they are used in the health care sector to monitor human physiological data (Verdone *et al.*, 2008). The following sections outline selected applications of wireless sensor networks.

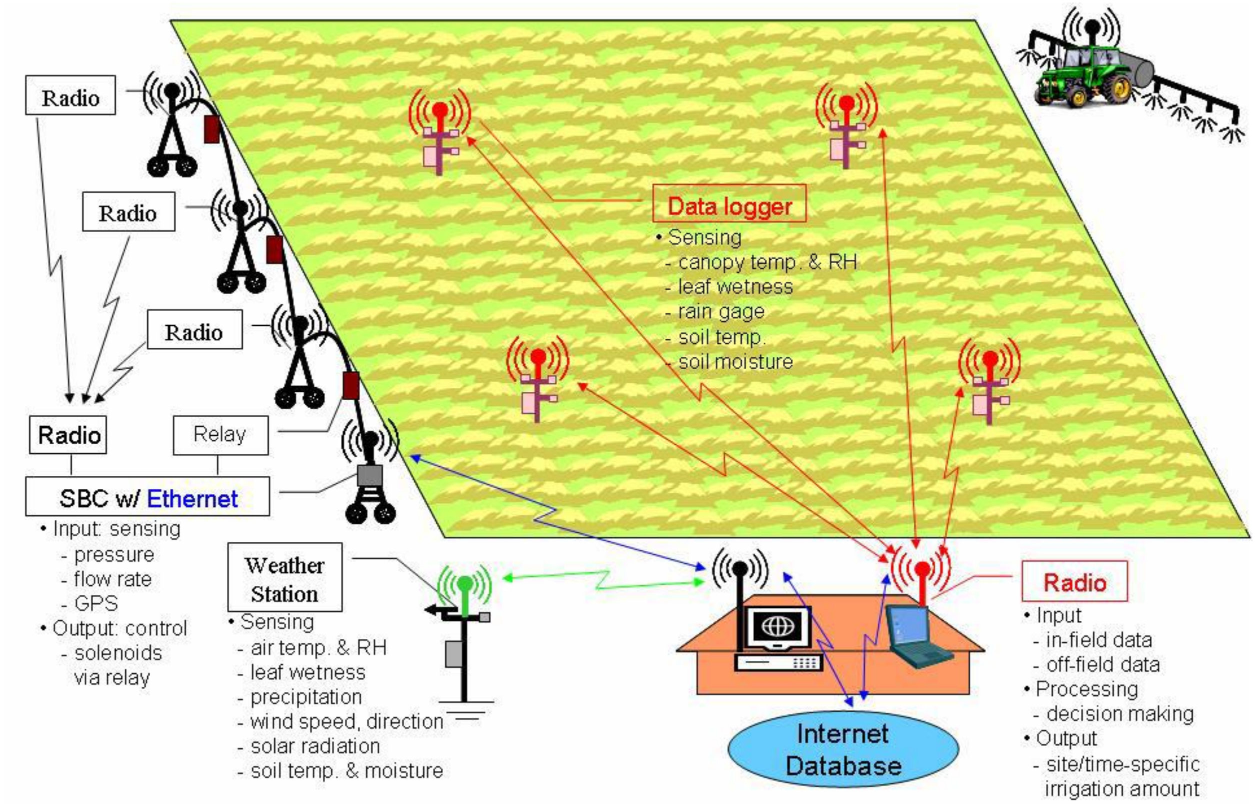


Fig 2.2 Conceptual layout of wireless network of in-field sensing stations for the real-time irrigation decision support system. (Kim et al 2006)

Precision agriculture and animal tracking

Sensors and sensor networks are important components of precision agriculture which aims at “maximum production efficiency with minimum environmental impact” (Taylor and Whelan, 2005). Land over-exploitation, one of the major concerns of intensive agriculture, leads to problems such as soil compaction, erosion, salinity and declining water quality (Wark *et al*., 2007). Sensors and sensor networks play a critical role in measuring and monitoring the health of the soil and water quality at various stages, from pre- to post-production. In the field of animal tracking, the movement of herds, the health of animals and the state of the pasture can be controlled via sensor networks. So far a number of sensor network systems have been developed and trials and field experiments are under way. However, concrete applications are at an early stage. This section briefly describes applications of sensor networks in precision agriculture and animal production. Subsequently, environmental impacts are presented qualitatively rather than quantitatively due to the early application stage. In precision agriculture, sensor networks can be used for:

- Plant/crop monitoring,
- Soil monitoring,
- Climate monitoring and
- Insect-disease-weed monitoring.

In the field of plant/crop monitoring, wireless sensors have been developed to gather, for example, data on leaf temperature, chlorophyll content and plant water status. Based on these data, farmers are able to detect problems at an early stage and implement real-time solutions. The health and moisture of soil is a basic prerequisite for efficient plant and crop cultivation. Sensors contribute to real-time monitoring of variables such as soil fertility, soil water availability and soil compaction. Further, sensor nodes which communicate with radio or mobile network weather stations provide climate and micro-climate data. Sensors registering the temperature and relative humidity can contribute to detect conditions under which disease infestation is likely to occur.

Wireless sensors are further used for precision irrigation, and systems developed for remotely controlled, automatic irrigation. Sensors assume, for example, the tasks of irrigation control and irrigation scheduling using sensed data together with additional information, *for example.* weather data (Evans and Bergman, 2003). Finally, sensors are used to assist in precision fertilization. Based on sensor data, decision support systems calculate the "optimal quantity and spread pattern for a fertilizer" (Wang *et al.*, 2006). Wireless sensor networks also contribute to a better understanding of the behavior of cattle, such as their grazing habits, herd behavior and the interaction with the surrounding environment (Wark *et al*., 2007). The information provided by these sensors helps farmers to understand the state of the pasture and to find optimal ways to use these resources. To test sensor applications for cattle management, Wark *et al*, (2007) attached sensor nodes to cattle collars. Sensors communicated in a peer-to-peer fashion. Cattle collars pinged each other "with each ping containing an animal's GPS position and time of each ping transmission" (Wark *et al.*, 2007). Based on the positioning data of each node and inertial information, the cattle's individual and herd behavior could be modeled and more general models could be developed. As a result, farmers are able to optimally manage environmental resources and plan grazing areas to prevent environmental problems such as overgrazing and land erosion. Current work focuses on the integration of sensor networks and radio frequency technology (RFID) as a significant number of cattle are equipped with RFID tags to record their ID as well as information such as cattle characteristics and food information.

Wireless sensors are also used for Environmental monitoring, urban terrain tracking and civil structure monitoring, Entertainment and Health care.

2.8 Challenges with WSN (tradeoffs)

Multi wireless communication

Long distance communication is only possible using prohibitively high transmission power, so use of intermediate nodes as relays can reduce the total required power.

Energy Efficient operations

Needed to support long lifetimes .There is need to look at energy efficient data transport between two nodes (J/Bit).

Auto Configuration

Nodes should be able to determine their geographical positions using only nodes of the network (self location) and also the network should be able to tolerate failing nodes or to integrate new nodes.

Collaborative and In-network processing

To solve the tasks of determining the average parameter for example temperature within an area and to report that value to a sink efficiently .Readings from individual sensors can be aggregated as they propagate through the network, reducing the amount of data to be transmitted and hence improving energy efficiency.

Data centric

Unlike traditional communication networks which are address centric in transfer of data between two devices WSN are concerned with values.

Power efficiency in WSNs is generally accomplished in three ways:

1. Low-duty-cycle operation.

2. Local/in-network processing to reduce data volume (and hence transmission time).

3. Multihop networking reduces the requirement for long-range transmission since signal path loss is an inverse exponent with range or distance. Each node in the sensor network can act as a repeater, thereby reducing the link range coverage required and, in turn, the transmission power. (Sohraby et al, 2007)

2.9 Classification of WSN

Design of a WSN is usually application oriented .As a result the architectures, protocols and algorithms of WSN vary case by case. (Yujin et al, 2010)

WSN can be classified according to distance of sensor nodes to the base station:

i. Single hop(non propagating)
ii. Multi hop (propagating)

In single hop all nodes transmit directly to the base station. While in multi hop WSN, some nodes deliver their data.

Factors	**Distinct Group**
Distance to base station/processing center	Single hop (non propagating) vs. Multi hop (propagating)
Data dependency	aggregating vs. Non –aggregating
Distribution of sensors	Deterministic vs. Dynamic
Control Scheme	Non Self configurable vs. Self configurable

Table 1.1

Advances in WSN

Advances in micro-electromechanical systems (MEMS) and continuous development in wireless communication are spurring more intelligent, less expensive, much smaller sensor nodes to be embedded into physical world for example. Pico node in Pico Radio project provide ubiquitous distribution of computation and communication for sensor /monitor networks. Each Pico node has a small size of less than 0.10 to 0.15 inch, consumes less than 10 mW and costs less than $1(Yujin et al, 2010).

Another system called WINS (wireless integrated networks sensor) integrate multiple functions including sensing ,signal processing ,decision making and wireless networking capability in a compact ,low power device. These intelligent sensors are tiny and powerful in establishing low cost and robust self-organizing networks for continuous sensing and event detection and identification.

A project called mAMPS (micro adaptive multidomain power aware sensors) has the objectives of implementing a micro sensor system on a chip of 1 cm, with integration of MEMS sensors, A/D, data and protocol processing and radio transceiver on a single die.

The Smart Dust project aims to explore the limits on size and power consumption of self-organizing sensor nodes that are not more than a few millimeters in size that is small enough to float in the air, detecting and communicating for hours or days (Villalba et al, 2009)

Sensor deployment Strategies

A fundamental issue for WSN .The objective of a sensor deployment plan is to achieve desirable coverage with a minimum number of sensor nodes while complying with constraints of QoS, cost, reliability and scalability of certain applications. In general four methods of sensor deployment exists in WSN that is.

Predetermined, Self regulated, randomly undetermined and biased distribution

Optimal Schemes at system level

Bandwidth efficient architectures and protocols can accelerate data delivery.

2.9.0 WSN Performance and Traffic Management

Topology management

Active topology of the network changes of time due to energy saving techniques of transmitting nodes to sleep or off states or node depletion. This leads to a critical issue of how to arrange sleep state transition while ensuring robust, un-degraded information collection.

A typical approach is to rotate the node functionality periodically to achieve balanced energy consumption among nodes. The protocol SPAN is an example of this approach for wireless ad hoc networks. Randomly a limited number of nodes are self selected as coordinates to construct the backbone in a peer to peer fashion within network for traffic forwarding while others make local decisions to transmit to sleep or keep active. Geographic Adaptive Fidelity (GAF) algorithm proposed in Xu is another way to rotate active nodes within a network .A new technique Sparse Topology and Energy Management (STEM) claims to improve beyond SPAN and GAF in terms of obtaining higher energy savings by trading off an increased latency to establish a multi-hop path.

Clustering and Hierarchical architecture

Energy consumed by communication is higher than that of sensing and computation therefore reducing amount of traffic and communication distance can greatly prolong system's life time. One approach is to divide entire system into distinct clusters which replaces one hop long distance transmission by multihop short distance data forwading.Various clustering techniques have been proposed Henzelman proposed a low energy adaptive clustering hierarchy technique. At the beginning each node self selects itself as a cluster head then advertises its decision to the other nodes which decide to join specific cluster that require minimum communication energy. In Power Efficient Gathering in Sensor Information Systems (PEGASIS), a chain based protocol, instead of sending data packets directly to cluster heads as with LEACH protocol, each node forwards its packets to the destination through the closest neighbor.

Collaborative Signal Information Processing (SCIP) and Data Aggregation

Local processing of data before direct forwarding will effectively reduce the amount of communication and improve the efficiency (information per bit transmitted).CSIP and data aggregation are two typical localized paradigm for data processing in WSN.CSIP is expected to provide solutions to many challenges including device spatial sampling of interested events, distributed asynchronous processing and communication, data fusion and querying and routing tasks. Data aggregation tries to minimize traffic load (number/length of packets) through eliminating redundancy. When intermediate node receives data from multiple source nodes, it checks the contents then combine them by eliminating redundant information under some accuracy constraints. There several data aggregation algorithms the most straight forward is duplicate suppression if multiple sources send the same data ,intermediate node will forward only one of them .Maximum and minimum functions are also very simple approaches.

2.9.1 WSN standard technologies

WSN vs. MANETS

There is also considerable research in the area of mobile ad hoc networks (MANETs). WSNs are similar to MANETs in some ways; for example, both involve multihop communications. However, the applications and technical requirements for the two systems are significantly different in several respects

The typical mode of communication in WSN is from multiple data sources to a data recipient or sink (somewhat like a reverse multicast) rather than communication between a pair of nodes. In other words, sensor nodes use primarily multicast or broadcast communication, whereas most MANETs are based on point-to-point communications. In most scenarios (applications) the sensors themselves are not mobile

(Although the sensed phenomena may be); this implies that the dynamics in the two types of networks are different. Because the data being collected by multiple sensors are based on common phenomena, there is potentially a degree of redundancy in the data being communicated by the various sources in WSNs; this is not generally the case in MANETs. Because the data being collected by multiple sensors are based on common phenomena, there is potentially some dependency on traffic event generation in WSNs, such that some typical random-access protocol models may be inadequate at the queuing-analysis level; this is generally not the case in MANETs. A critical resource constraint in WSNs is energy; this is not always the case in MANETs, where the communicating devices handled by human users can be replaced or recharged relatively often. The scale of WSNs and the necessity for unattended operation for periods reaching weeks or months implies that energy resources have to be managed very judiciously. This, in turn, precludes high-data-rate transmission. The number of sensor nodes in a sensor network can be several orders of magnitude higher than the nodes in a MANET.

For these reasons the plethora of routing protocols that have been proposed for MANETs are not suitable for WSNs, and alternative approaches are required.(Sohraby et al ,2007)

Currently three standard technologies are available for WSN: **ZigBee, Bluetooth and Ultra-wideband (UWB).** All of which operate within the Industrial Scientific and Medical (ISM) band of 2.4 GHz, which provides license free operations, huge spectrum allocation and worldwide compatibility. In general, as frequency increases, bandwidth increases allowing for higher data rates but power requirements are also higher and transmission distance is considerably shorter .Multi-hop communication over the ISM band might well be possible in WSN since it consumes less power than traditional single hop communication.

It is also possible to create a WSN using Wi-Fi (IEEE 802.11), but this protocol is usually utilized in PC-based systems because it was developed to extend or substitute for a wired LAN. Its power consumption is rather high.

Bluetooth (IEEE 802.15.1) was developed as a wireless protocol for short-range communication in wireless personal area networks (PAN) as a cable replacement for mobile devices. It uses the 868 and915 MHz and the 2.4 GHz radio bands to communicate at 1 Mb per second between up to seven devices (Ruiz et al 2009).Bluetooth is mainly designed to maximize ad hoc networking functionality. Some of its common functions are passing and synchronizing data, for example. between a PDA (personal digital assistant) and a computer, wireless access to LANs, and connection to the internet. It uses frequency-hopping spread-spectrum (FHSS) communication, which transmits data over different frequencies at different time intervals. Bluetooth uses a master-slave-based MAC (medium access control) protocol. Bluetooth has a transmission range of 8-100m,can support up to 8 nodes, can support audio, pictures, graphics and files data types, it has a data rate of 1 mb/s and a latency of < 10s ,battery life of 1 week and the network is in extendable.

The **ZigBee** standard is built on top of the IEEE 802.15.4 standard. The IEEE 802.15.4 standard defines the physical and MAC (Medium Access Control) layers for low-rate wireless personal area networks. The physical layer supports three frequency bands with different gross data rate 2,450 MHz (250 kbs-1), 915 MHz (40 kbs-1) and 868 MHz (20 kbs-1). It also supports functionalities for channel selection, link quality estimation, energy measurement and clear channel assessment. It uses Direct Sequence Spread Spectrum (DSSS) communication which transmits data in sequence. The ZigBee alliance was formed because its members felt that existing standard technologies were not applicable to ultra-low power application scenarios.

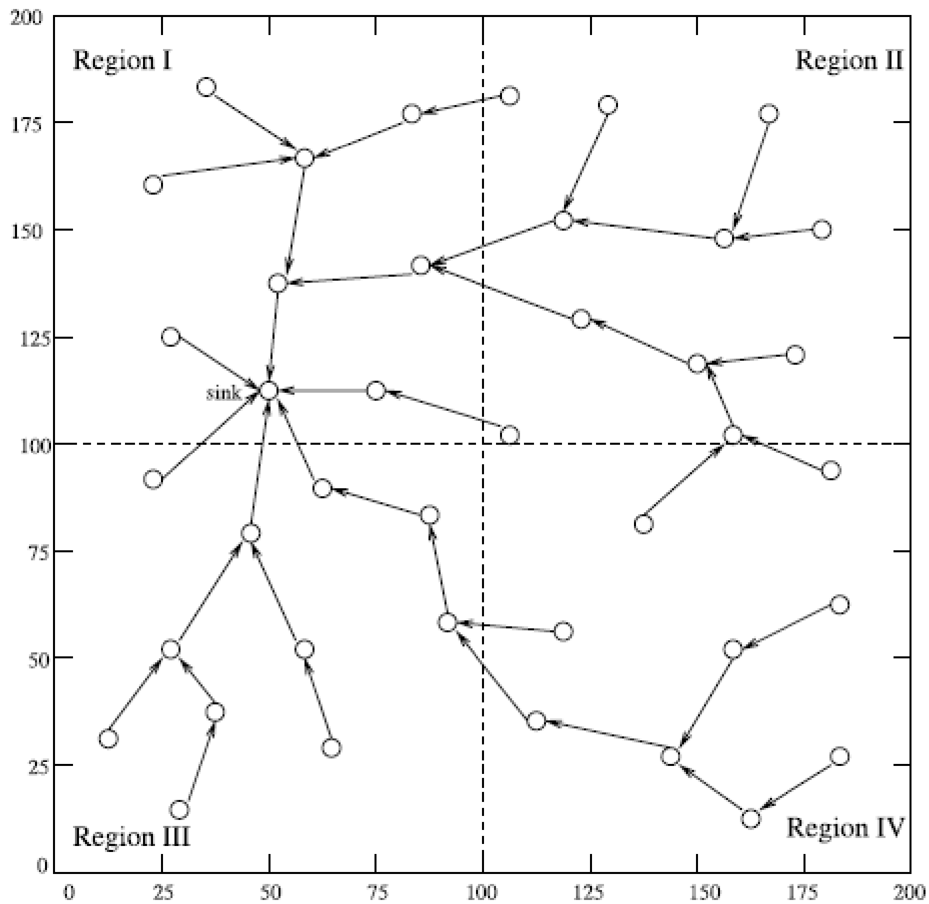
ZigBee standardizes both the network and the application layer. The network layer is in charge of organizing and providing routing over a multi-hop network, specifying different network topologies: star, tree, peer-to-peer and mesh. The application layer provides a framework for distributed application development and communication. Zigbee has a transmission range of 1-100m,can support up to 65 000 nodes, can support data packets data types, it has a data rate of 20-250 Mb/s and a latency of 30s ,battery life of >1 year and the network is extendable(Ruiz et al 2009).

Bluetooth supports a wider range of traffic types than ZigBee however it consumes more power than Zig Bee which was for low power consumption.
Zig Bee provides higher network flexibility, allowing different topologies .Zig Bee allow a larger number of nodes more than 65 000 () according to specification. Thus Zig Bee has been approved by various authors for monitoring in agriculture.
Ultra-wideband (UWB) is a technology for transmitting large amounts of data over a wide spectrum of frequency bands with low complexity, low cost, and low power consumption. This technology has an enhanced capability to penetrate through obstacles. USB Dongle WUWBD-101 is designed based on the UWB technology and is available for the market .Other short-range methods, including Crossbow's MICA2, provide high-level functional integration designed to extend the wireless mesh networking technology into a wide variety of sensing applications. An example is a multi-hop wireless sensor network using MICA2 to monitor wildfire behavior changes due to temperature, relative humidity and wind (Wei Han, 2011).

2.10 Routing Protocols

Routing techniques are needed to send data between sensor nodes and the base station. Several routing protocols are proposed for sensor networks. These protocols can be divided
Into the following categories: data-centric protocols, hierarchical protocols, location based protocols, and some Quos-aware protocols. (Barati et al, 2008).Data centric can be further divided into event driven and query driven examples being SPIN (Sensor Protocols for Information via Negotiation) , directed diffusion , Gradient Broadcast (GRAB), and Rumor

routing. Hierarchical protocols include LEACH (Low-Energy Adaptive Clustering Hierarchy), Threshold-sensitive Energy Efficient sensor Network protocol (TEEN) (Wei Han, 2011) and Adaptive Periodic TEEN (APTEEN) are the follow-up work of LEACH. TEEN is designed to be responsive to sudden changes in the sensed attributes. (Barati et al ,2008)Yujin et al ,2010) Proposed a data forwarding protocol, which they designed and tested under simulation using NS2 and in different sized network topologies and confirmed that it was reliable.



0

Fig 2.3 an exemplary network routing tree for 40 nodes placed in a 200x200 area. (Sungare et al, 2008)

Transport Protocols

TCP and UDP are the popular transport protocols used in networks but they have shortcomings when applied to WSN,Some of the problems faced includes:

Lack of interaction between TCP or UDP and the lower layer protocols which are important in WSN as they provide important information to the transport which enhances system performance.TCP’s three way hand shake technique is a large overhead for small volumes of event driven data from the sensors

UDP is not energy efficient for WSN because it lacks flow and congestion control mechanism.

Examples of available Transport Protocols for WSN

CODA (Congestion Detection and Avoidance)

CODA detects congestion by monitoring the buffer occupancy and wireless channel load .if the occupancy exceed threshold then it implies that congestion has occurred. (Sohraby et al, 2007)

ESRT (Event-to-Sink Reliable Transport)

ESRT uses end to end approach to guarantee a desired reliability. It periodically computes a reliability figure(r), representing the rate of packets received successfully in a given time interval. ESRT then deduces the required sensor reporting frequency (f) from the reliability figure (r) using an expression such as f=G(r) Finally, ESRT informs all sensors of the values of (f) through an assumed channel with high power.

RMST (Reliable Multisegment Transport)

RMST guarantees successful transmission of packets in the upstream direction. Intermediate nodes cache each packet to enable hop-by-hop recovery, or they operate in non cache mode, where only end hosts cache the transmitted packets for end-to-end recovery. RMST supports both cache and non cache modes. Furthermore, RMST uses selective NACK and timer-driven mechanisms for loss detection and notification. In the cache mode, lost packets are recovered hop by hop through the intermediate sensor nodes. If an intermediate node fails to locate the lost packet, or if the intermediate node works in non cache mode, it will forward the NACK upstream toward the source node. RMTS is designed to run above directed diffusion which is a routing protocol, in order to provide guaranteed reliability for applications. Problems with RMST are lack of congestion control, energy efficiency, and application-level reliability. (Sohraby et al, 2007)

PSFQ (Pump Slowly Fetch Quickly)

PSFQ Distributes data from sink to sensors by pacing data at a relatively slow speed but allowing sensor nodes that experience data loss to recover any missing segments from immediate neighbors. This approach belongs to the group downstream reliability guarantee. The motivation is to achieve loose delay bounds while minimizing loss recovery by localizing data recovery among immediate neighbors. PSFQ consists of three operations: pump, fetch, and report. This is how PSFQ works: Sink broadcasts a packet to its neighbors every T time units until all the data fragments have been sent out. Once a sequence number gap is detected, the sensor node goes into fetch mode and issues a NACK in the reverse path to recover the missing fragment. The NACK is not relayed by the neighbor nodes unless the number of times that the NACK is sent exceeds a predefined threshold [7.4]. Finally, the sink can ask sensors to provide it with the data delivery status information through a simple and scalable hop-by-hop report mechanism. PSFQ has the following disadvantages: It cannot detect packet loss for single packet transmission; it uses a slow pump, which results in a large delay; and hop-by-hop recovery with cache necessitates larger buffer sizes.

GARUDA

GARUDA uses a NACK message for loss detection and notification. Loss recovery is performed in two categories: loss recovery among core sensor nodes and loss recovery between noncore sensor nodes and their core node. (Sohraby et al, 2007)

ATP (Ad Hoc Transport Protocol)

ATP is based on a receiver and network-assisted end-to-end feedback control algorithm. It uses selective ACKs (SACKs) for packet loss recovery. In ATP, intermediate network nodes compute the sum of exponentially distributed packet queuing and transmission delay, called D. The required end-to-end rate is set as the inverse of D. The values of D are computed over all packets that traverse a given sensor node, and if it exceeds the value that is piggybacked in each outgoing packet, it updates the field before forwarding the packet. The receiver calculates the required end-to-end rate (inverse of D) and feeds it back to the sender. Thus, the sender can intelligently adjust its sending rate according to the value received from the receiver. To guarantee reliability, ATP uses selective ACKs (SACKs) as an end-to-end mechanism for loss detection. ATP decouples congestion control from reliability and as a result, achieves better fairness and higher throughput than TCP. However, energy issues are not considered for this design, which raises the question of optimality of ATP for an end-to-end control scheme.

MAC Protocols

MAC protocols affect the efficiency and reliability of hop-by-hop data transmission. Existing MAC protocols such as the IEEE 802 series standard may not be completely suitable for WSNs because of energy efficiency. General MAC protocols can result in a waste of energy.

Data Aggregation Protocols and Techniques

Existing Systems

- A wireless solution for intelligent field irrigation system dedicated to Jew's ear planting was developed in Lisui,Zhejiang ,China in 2009.It was based on Zig Bee technology but was not implemented on large scale .It appeared in Wireless Communication and trusted computing ,2009 NSWCTC '09 International conference on Network Security.
- DFeliciano,Cayanan,M Dixon,Y Zheng from University of Guelpha developed a conceptual model of an automated irrigation system in 2010.They developed a prototype automated irrigation system using wireless modules and in situ root zone soil moistures, capacitance sensors ,EC and temperature sensors .The wireless sensors were deployed throughout the greenhouse and root zone data was transmitted to a computer control system
- (Balendonk et al 2009) designed a model wireless sensor based system with 6 SM200 soil moisture sensors ,3 repeaters and gateway/base station connected to a PC and Mesh topology which support multipath communication hence more reliable was used. The nodes were able to relay data to a repeater over 20m distance. But the desired maximum data loss of 5% could not be fulfilled. Battery life, remote access and internet data transport worked well .The system's weak points were signal losses, sensor performance, high cost and packaging.
- Delta T Devices (UK), Netafim (IS), Decagon (US) and Crossbow (US) are among the major suppliers of wireless sensor equipment and were very active in the WSN research

.However the equipment is still expensive and uses a lot of energy to overcome the variable damping of electromagnetic waves in crops under fluctuating weather conditions (Balendonk et al 2009).

(Zhang, 2004) proposed a WSN for precision agriculture using Bluetooth. Although challenges such as battery life and transmission latency exist in his application, his work gives hopes for the future of WSN in agriculture applications. (Wei Han, 2011)

(Abhinav et al, 2010)Designed a protocol which they named Distributed Sensor Webs Routing Protocol (DSRP) and a WSN system which they implemented to monitor water status and control irrigation for ornamental crops. However the system was developed for compatibility with EM50 data loggers of Decagon Devices Inc which poses a question of compatibility with other devices from different vendors.

Most research about the use of WSN in the field of precision agriculture and horticulture are so far carried out in Australia and North America.

A number of publications confirm that at the current stage WSN are not reliable enough, can't stand outdoor climate conditions, lose communication, are not fault tolerant and uses too much power despite the fact that a lot of researches have been carried out to address these different issues. This unreliability is caused by many factors which range from the sensor hardware, software, network infrastructure, protocols and if it's in precision irrigation where data about the field is needed for scheduling and decision making , we see more dependency on the data collection that is. how efficient and accurate is the data collected. Thus our main thrust for this research to focus on efficient data collection for effective irrigation management and water conservation as well as improving on our yield in the long run.

(Van Lersel et al ,2010) also confirmed that although automating irrigation is easy, automated systems are not necessarily water efficient I tend to agree with this statement, for automation is fully dependant on the collected data from the sensors, and there are many factors which can affect the efficient collection of this data which starts from the sensor itself, the network or transmission medium, necessary calculation and processing of the data, placement of the sensor nodes in the network and so on.

Since battery powered equipment are more favorable, there is need for both equipment and communication protocols improvement so as to conserve energy and increase reliability under outdoor agricultural conditions.

Since we are looking forward to implementation of these systems in large scale agriculture, for us to get more accurate readings and hence more precise irrigation we need to subdivide a field into regions, take note of the soil type and any relevant data that might help us in our decision making during scheduling for example. FLOW AID made a Decision Support System(DSS) and they used this method .They divided the land into plots and then measured amount of water used against soil type, water availability and yield. They used the 866-868 MHz frequency band for the sensors. So as control experiment we would propose to use a manual or time driven scheduling to test against our system in order to see the tradeoffs.

2.11 Simulation and Simulators

Since I will have to run some simulations in our study, we will look at the available simulators and ascertain the best choice. Generally there are many simulators for example.

- NS, PDNS (Parallel/Distributed NS).
  "The PADS research group at Georgia Tech has developed extensions and enhancements to the ns simulator to allow a network simulation to be run in a parallel and distributed fashion, on a network of workstations." Last supported for ns-2.27 (January 2004).

- Georgia Tech's Dynamic Network Emulation Backplane Project.
  "The backplane enables the user/modeler to bring multiple network simulators together and harness their models in a single experiment... The backplane also supports incorporation of actual network applications into the execution, to execute over the emulated network."

- GloMoSim/Parsec.
  " GloMoSim built a scalable simulation environment for wireless and wired network systems. It was designed using the parallel discrete-event simulation capability provided by Parsec."
  - QualNet: QualNet was a commercial simulator that grew out of GloMoSim.

- SSF (Scalable Simulation Framework).
  It includes SSF Network Models (SSFNet), with "open-source Java models of protocols (IP, TCP, UDP, BGP4, OSPF, and others), network elements (hosts, routers, links, LANs), and

assorted support classes for realistic multi-protocol, multi-domain Internet modeling and simulation", and a networks with models of large, realistic BGP topologies, heavily congested networks, and validation models, together with model descriptions and associated publications.

- Dartmouth SSF (DaSSF).

  "Dartmouth SSF (DaSSF) is a process-oriented, conservatively synchronized parallel simulator, which is designed for but not exclusively for simulating very large scale multi-protocol communication networks. DaSSF is a C++ implementation of Scalable Simulation Framework (SSF)."

- GTnetS

  "The Georgia Tech Network Simulator (GTnetS) is a full-featured network simulation environment that allows researchers in computer networks to study the behavior of moderate to large scale networks."

- JavaSim now renamed to J-Sim.

- OMNET++.

  OMNET++ is free for academic and non-profit use, and contains IP, IPv6, MPLS, mobility, and ad-hoc simulations.

- The M5 Simulator.

  M5 simulates TCP/IP performance with full-system support with detailed I/O models and three CPU models.

- HEGONS.

  "HEGONS is a Heterogeneous Grooming Optical Network Simulator that supports mixed routing and wavelength assignment algorithms and optional wavelength conversions capability on each node. The goal of Hegons is the evaluation of different dynamic routing and wavelength assignment (RWA) algorithms in WDM optical networks."

Peer-to-peer simulators:

- The State of Peer-to-Peer Simulators and Simulations,

  S. Naicken B. Livingston A. Basu S. Rodhetbhai I. Wakeman D. Chalmers CCR, April 2007.

- Tools for Peer-to-Peer Network Simulation,
  internet-draft draft-irtf-p2prg-core-simulators, work in progress.

Commercial simulators:

- OpNet Modeler.
  OpNet Modeler is a leading commercial network simulator, including a "library of detailed protocol and application models including Multi-Tier Applications, Voice, HTTP, TCP, IP, OSPF, BGP, EIGRP, RIP, RSVP, Frame Relay, FDDI, Ethernet, ATM, 802.11 Wireless LANs, MPLS, PNNI, DOCSIS, UMTS, IP Multicast, Circuit Switch and many more... The Standard Model Library includes hundreds of vendor specific and generic device models including routers, switches, workstations, and packet generators".

- Omnicor's Net Disturb, IP network emulator software to generate impairments over IP networks.

- QualNetis a high-fidelity network simulator based on GloMoSim. It uses a parallel simulation engine to run large wireless and wired networks.

- **Other commercial simulators:**
  Netwiser (including a network simulator);
  Shunra (for performance testing for applications).
  NetScale, a scalable network simulation tool based on patented mathematical developments, from a start-up company stemming from INRIA.
  NetSim is a commercial network simulator for use at the undergraduate level, mostly at use in India. (icir.org ,accessed 15/102011)

Of all these I have chosen OMNETT 4.2.1, which is an open source and recent simulator developed to address the shortcomings of NS2. Ns-3 is popular simulation tools for network simulation for academic non-profit purposes. The tool is written in c++ and runs on an Eclipse IDE. (There is support for many types of 802.11 networks including 802.11s mesh networks).

CHAPTER 3

3.0 METHODOLOGY

The overall goal of this research is to realize important factors which matters in real application like: Advantages of automatic pump and valves, potential water, energy and money saving realized by efficient precision irrigation which as I proposed is dependent on the accuracy of the data collected .Not only that but also on the effectiveness since this is real time data they should not be delays in data propagation since data expires and sometimes you might miss important events that needs urgent attention and has direct impact on the plant growth.

Precision irrigation falls into two categories that is. one used for gathering environ mental data and one used for automatically controlling the irrigation system. Environmental data is used to determine crop water requirement. Formulas are there which incorporate different environmental parameters to determine the amount of water lost by the plant. Our quest is to accurately determine this value in real-time and be able to replace only the deficit water there by maintaining the water levels needed by the plant for optimum growth and yields.

To achieve this we may take advantage of supporting technologies in form of WSN, internet connections, switching hubs, routers and gateways for transmission of sensed data which will be used for calculating water thresholds and mark a range of acceptable levels and to determine critical values. You can also incorporate meteorological data from Metrological data collection systems to make your calculation more accurate.

In order to achieve more accurate calculation of the evapotranspiration threshold and plant water loss, we need to consider as many environmental, plant and climatic factors.eg soil moisture, temperature, humidity, leaf surface evaporation, plant diameter, solar radiation and so on.

Field environment moisture sensors are the most common type of environmental sensors and since this an experimental study we will not include all parameters for now but focus on soil moisture as a major deciding factor on the plant water loss that is. we will be focusing on maintaining the soil water levels which are best for the plant growth .Sensors are strategically located at a number of points within the crop field in a way that covers variation in soil type and climate. This is done to achieve variable irrigation scheduling across the field to get more effective water replacement. There are different topologies and node placement algorithm. I investigated the effectiveness of the most common ones using the OMNeT ++ simulator.

We explain here the concept used in this research as self contained, that is, wireless sensor nodes that form an ad hoc network to relay data back to a central location. Data gathering protocols are formulated for configuring the network and collecting information from the desired environment. In each round of the data gathering protocol, data from the nodes need to be collected and transmitted to Base Station, where from the end user can access the data. A simple way of doing that is aggregating (sum, average, min, max, count) the data originating from different nodes. A more elegant solution is data fusion which can be defined as combination of several unreliable data measurements to produce a more accurate signal by enhancing the common signal and reducing the uncorrelated noise. Sensor nodes use different data aggregation techniques to achieve energy efficiency. The aim is efficient transmission of all the data to the base station so that the lifetime of the network is maximized in terms of rounds, where a round is defined as the process of gathering all the data from sensor nodes to the base station, regardless of how much time it takes. Existing data gathering protocol can be classified into different categories based on the network structure and protocol operation based on routing protocols that aim at power-saving and prolonging network lifetime are intensively studied in research community.

3.1Placement of wireless sensor node for relatively effective field coverage

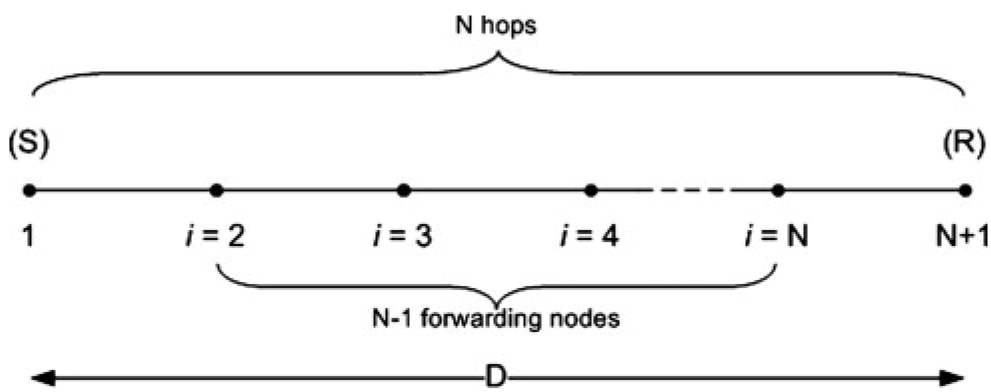
Optimum placement whether dynamically or statically has equal advantages of maximizing network lifetime, improving network efficiency, reduce number of sensors to be used and increase the coverage thereby improving data collection which in turn would have a direct impact on the level of precision in our precision irrigation. While many published papers aim at maximizing the lifetime our aim is to maximize utilization efficiency and coverage for optimum data collection. Yunxia Chen et al 2009,(Al-Karaki et al 2010) proposed a new performance metric, called lifetime per unit cost, to measure the utilization efficiency of sensors. They defined the lifetime per unit cost as the network lifetime divided by the number of deployed sensors. They found that deploying either an extremely large or an extremely small number of sensors leads to low lifetime per unit cost. We are thus motivated to optimize both the number of sensors and their placement and data aggregation for increased lifetime, coverage and efficiency.

We find that sensors should be placed more uniformly as their sensing range or the path loss exponent increases, and more sensors should be deployed as the event arrival rate increases or the sensing power consumption decreases.

Sensors monitor the events of interest and send them to the gateway node where the end user can access it. Due to power limits and hardware constraints every sensor has a sensing range of r km and a communication range of 2r km.Sensor placement is according to their distance from the gateway node.

Let S_1-------S_N be the number of sensor where $S_1$is the closest to the gateway node and S_N is furthest from the gateway node.S_i is the i-th sensor from the gateway node. Sensor placement $\{d_i\}_{I=1}^{N}$ according to distance between adjacent sensors d_i should satisfy the following constraints:

(1) $0 \leq d_1 \leq$r
(2) $0 \leq d_i \leq 2$r
(3) for $2 \leq i \leq N-1$
(4) $0 < L - \sum_{j=1}^{N} d_j < r$



(4)

Fig 3.0 Node placement (Al-Karaki et al 2010)

When an event occur or the sensor node is queried for data, the sensor node that is closest to the event initiates the reporting process by generating an equal sized packet send it to the nearest adjacent neighbor. The sensed signal strength decreases with the sensing distance so the sensor with the strongest sensed signal and hence closest to the event will initiate the reporting process. We employ greedy sensor placement against uniform sensor placement where sensors are equally spaced.

3.2 Experiment setup

Using OMNeT 4.2.1 simulator, different topologies were simulated taking note of their effect on the network lifetime and also delays in packet propagation between nodes as data is transmitted to the sink. This data is then used to determine the best node placement and topology based on the improved lifetime and reduced delays in packet propagation between hops.

Assumptions: we assume that routing technique used is no-trivial and the network has no other constrains affecting it like transmit power and strength of the sensors.

Parameters:

Network lifetime: A measure of the expected energy dissipation rate which determine how long the network will run perfectly before the nodes run out of power. Since transmit power and radius and distance between the communicating nodes determine how much energy is needed for each and every transaction the node send the packets, this parameter is relevant to calculate the efficiency of the whole network. Low lifetime implies the network in inefficient that is. there is high communication cost.

Network Performance: measured as the number of events processed per second. It will be used also to evaluate the topology's efficiency.

After setting up all the necessary infrastructure and configuration, Sensors may be queried manually or automatically by a data collection system. Automatic data collection will query at intervals (15 minutes or so) then log data into a database for subsequent reference.

Automatic data collection system require a communication network of low cost data collection nodes .Any node in the network may have one to several sensors attached. Some nodes can be used as relay nodes in the Wireless Network. Viewing real-time data as well as database archive may be limited to a local network on the farm or may be accessed on the internet.

Automatic controls provide more accurate and reliable scheduler than human operators. In principle the amount of time needed for an irrigation system to run is based on the amount of water crops need for a particular application. The time is either calculated formally based on known or estimated water application rates. For automatic control, sensors are also needed at the valve control points for example, pressure transducers used to detect overpressure at the pump discharge. Other sensors that can be used include flow rate, flow total, well water level

Using SMTP protocols message can be delivered as text messages on cell phones.

3.3 Irrigation System Design

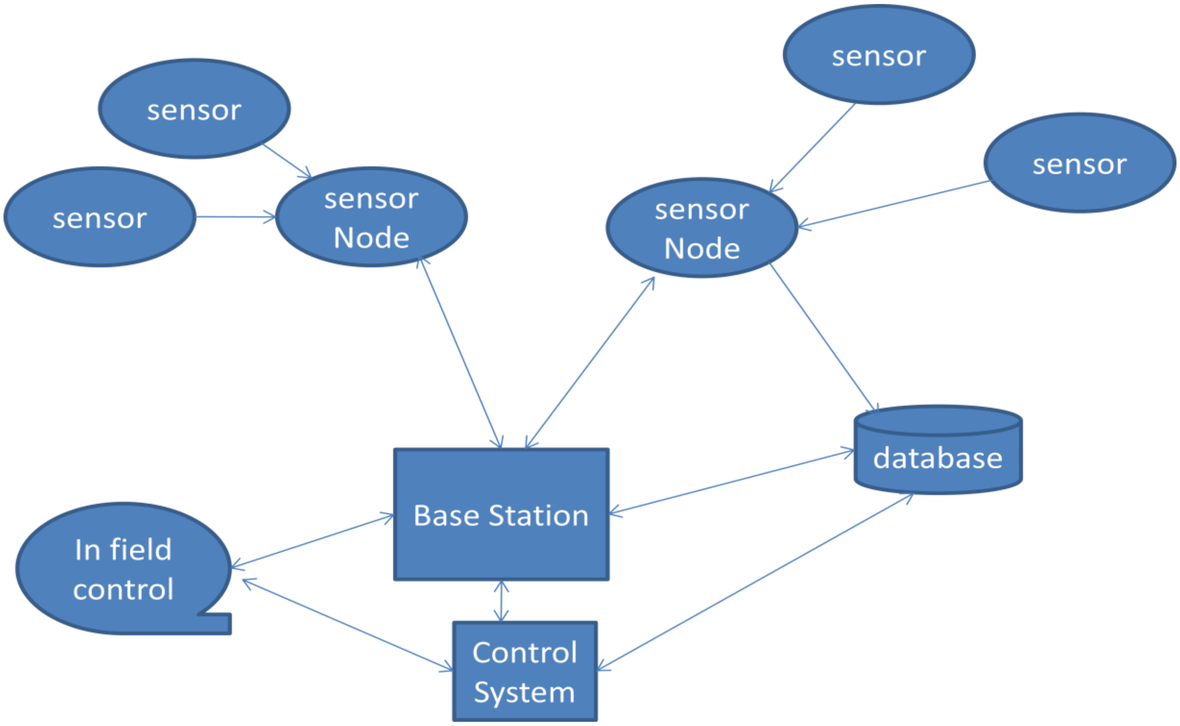


Fig 3.1Clustering Conceptual Model

The total number of sensor nodes and actuators are depends on the size of greenhouse. About 200 nodes are sufficient if the size of green house is 35m x 200 m.The sensor nodes can be classified as ‘A’ and ‘B’, where type ‘A’ is climate sensor for sensing climatic factors like Temp, Pressure, Light, Humidity, CO2, Wind speed and wind direction. Sensors can be placed at a distance of 10 to 15 meters of diameter, to capture precise environmental condition. The type ‘B’ sensors are soil sensors, which are recommended to use, as per the layout plan of the crop plantation. They can also control water flow of irrigation system used in the field. They are typically used after every two meters. (Waghmare et al, 2011)

Different crops have different parametric value ranges which are acceptable and these are fed into the system as controls for example:

Crop	Temp °C	CO_2 PPM	Light K Lux	Moisture		pH Value
				Air	Soil	
Carnation	16-22	1000	45-50	65	16	5.5-7.0
Gerberas	27-30	1000	35-40	65	17	5.5-6.5
Anthurium	24-26	1000	18-35	75	20	5.5-6.5
Tomato	16-35	1500	45-50	65	16	5.5-7.0
Roses	15-30	1000	30-40	70	17	4.0-5.5

Table 2.1 Input parameter for the System (Ghosh et al, 2010)

Node Placement Designs

Topology1

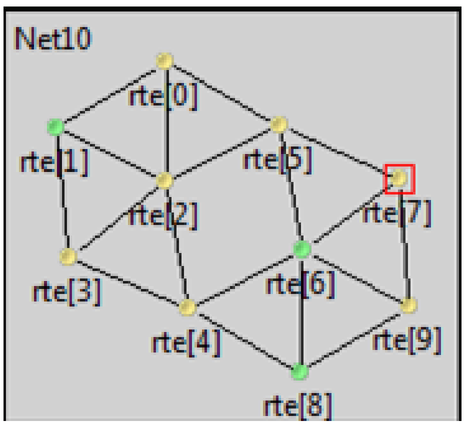


Fig 3.2

Topology 2

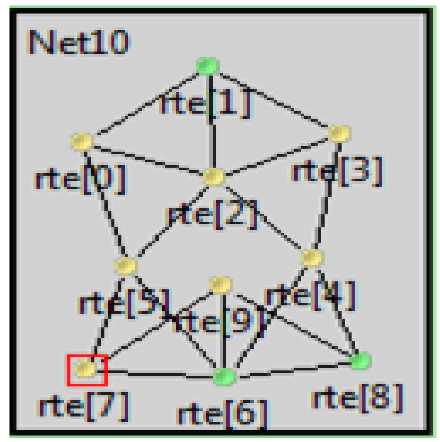


Fig 3.3

Topology 3

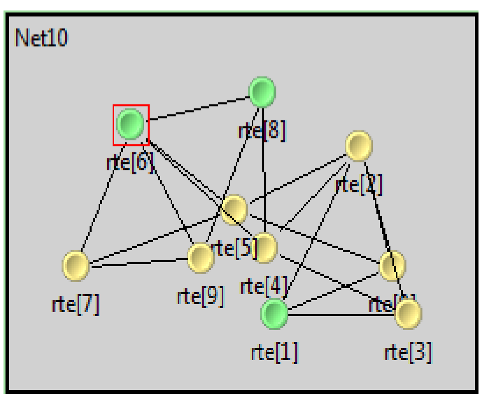


Fig 3.4

Topology4

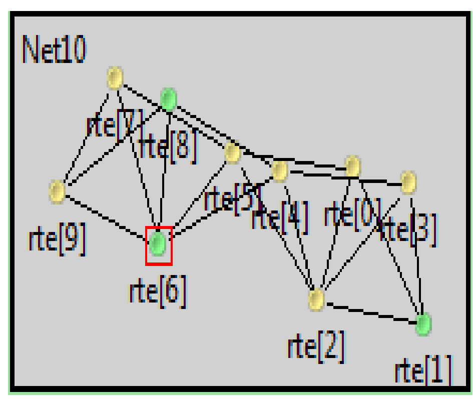


Fig 3.5

Star

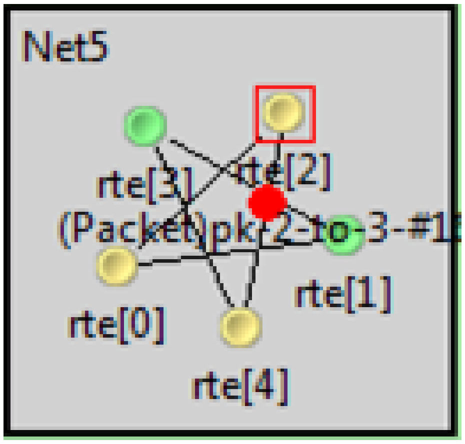


Fig 3.6

Pentagonal

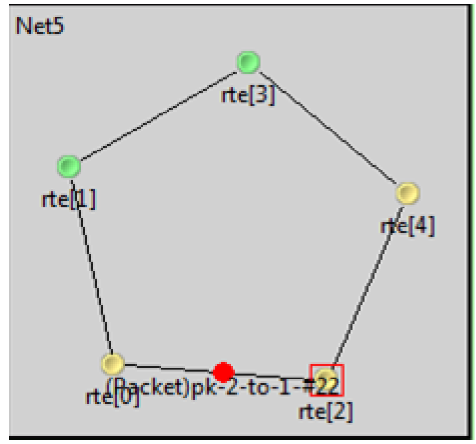


Fig 3.7

Kite

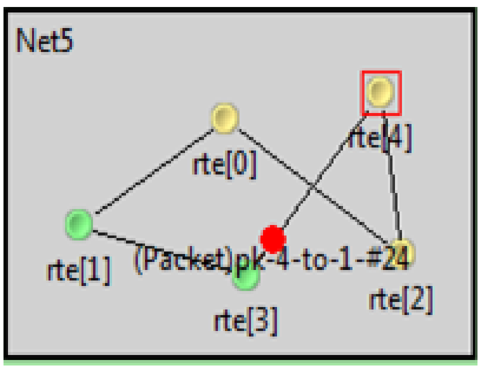


Fig 3.8

Linear

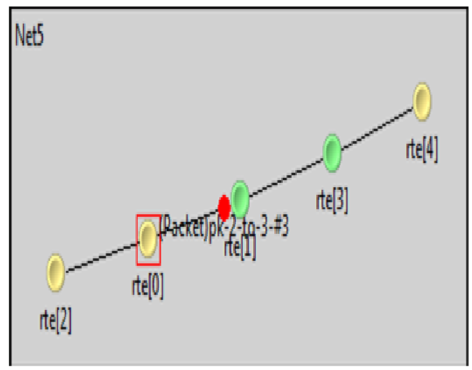


Fig 3.9

Mesh

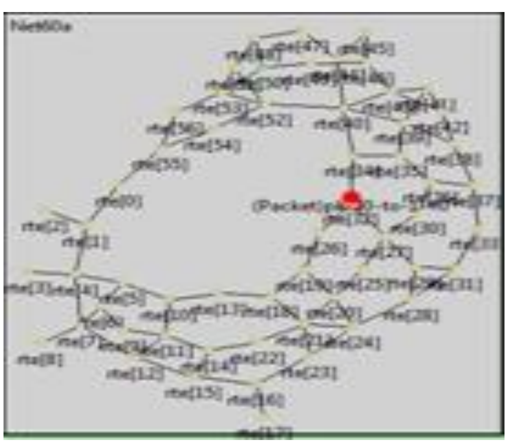


Fig 3.1.0

Ring

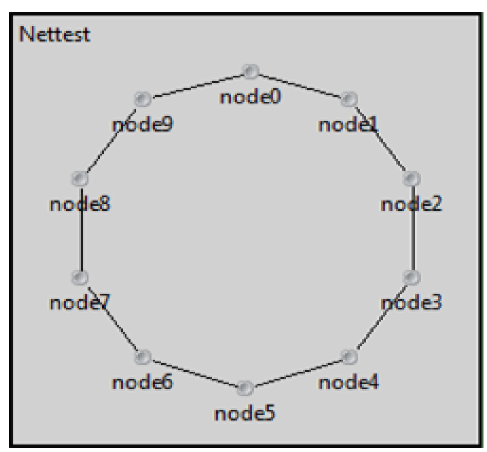


Fig 3.1.1

3.4 Data Aggregation techniques

Because of power and transmission range limitations, data dissemination in sensor networks is typically carried out as a collective operation, in which sensors collaborate to get data from different parts of the sensor network to the information sinks. One way of performing power-efficient data collection in sensor networks is to process the data as it flows from information sources to sinks. This technique is commonly referred to as (in-network) data aggregation and can be quite effective at conserving power. (Chong et al, 2008)

- Greedy Aggregation (GA) is based on an energy efficient tree constructed by connecting each source to the closest point of the existing tree to improve energy efficiency. However, Greedy Aggregation incurs additional costs for the generation and propagation of incremental cost messages yet still it may not find the most energy efficient path. Motivated by the limitations of the existing GA algorithm, Enhanced Greedy Aggregation (EGA) algorithm for Directed Diffusion Protocol (DDP) was developed to encourage multiple-path sharing. Evaluations of the performance of EGA compared to the original DDP shows a significant reduction in average dissipated energy. Investigation was done on the impact of aggregation delay, number of shared paths, task duration overlap, and node density on average dissipated energy and average delay.

- Data aggregation tries to minimize traffic load (number/length of packets) through eliminating redundancy. When intermediate node receives data from multiple source nodes, it checks the contents then combine them by eliminating redundant information under some accuracy constraints. There several data aggregation algorithms the straightest forward is duplicate suppression if multiple sources send the same data, intermediate node will forward only one of them.

Duplicate suppression Algorithm

- T_1 get data from neighboring nodes(labeled)
- T_2 Compare data
- If (duplicate)
- Take one and discard the rest
- T_3 calculate the average of this data

- T_4 Forward the averaged data to the next sink on request or at time intervals

Data aggregation

The main idea of data aggregation and in-network processing techniques is to combine data arriving from different sources (sensor nodes) at certain aggregation points (aggregators) en route, eliminate redundancies by performing simple processing at the aggregation points and minimizing the total amount of data transmission before forwarding to the Base station. Removing redundancies results in transmitting fewer numbers of bits hence reduces energy consumption and increases node's lifetime. Data aggregation is the heart of data centric routing against the traditional address centric routing which just implements the shortest path paradigm.

Directed Diffusion

In our experiment we incorporate the directed diffusion paradigm. Directed diffusion is a data centric data communication mechanism for wireless sensor networks where by data sources and sinks use attributes to identify what information they need or provide.

The sink broadcast its interest for certain information to all the nodes, then the nodes with the desired information will set a gradient or best path to the sink then data is propagated from source to sink. This mechanism is useful in our case when we want to query data periodically. The base station will send these queries at 15 min intervals and the data will be propagated, being aggregated along the way in the intermediate aggregate nodes.

There are three scenarios in the way data is transmitted from different sources.

I. All sources send completely different information (no redundancy).
II. All sources send identical information (complete redundancy).
III. All sources send information with some intermediate, non-deterministic, level of redundancy.

The first case, no data aggregation is needed since it all brings out same results either ways. Second case and third cases, data aggregation shows its greater advantage since some in-network processing based on the data from the sources is needed to eliminate redundancy, minimize unnecessary transmission of data and saving transmission energy.

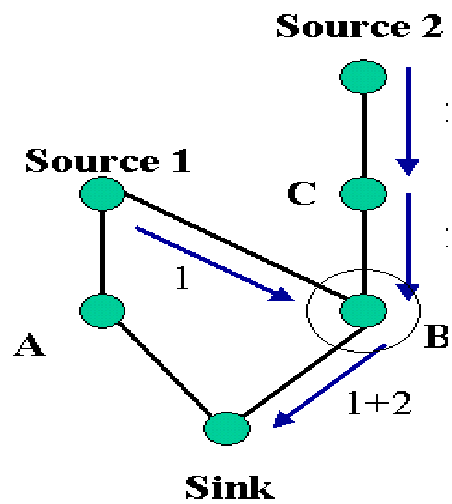


Fig 3.1.2 Source to Sink Packet Forwarding (Bhaskar et al, 2010)

Of the data aggregation function Max, Duplicate Suppression, Average and Min the simplest function is duplicate suppression. If sources 1 and 2 both sends the same data, then node B will only forward one of these data to the sink. To simplify modeling our assumption is that the aggregator node transmits only one aggregate packet even if it receives data from multiple sources.

Let's say we have n sources labeled S_1 …..S_n and a sink R.Let the network graph G= (V, E) consists of all nodes V with E made of all edges between nodes that came directly communicate to each other. Assuming that the number of transmissions of any node in the data aggregation is exactly one, the data aggregation tree comes out like the reverse of a multi-cast tree. Instead of a single source sending packets to all receivers, all sources are sending data to the same sink. It is common knowledge that a multicast tree is a minimum Steiner tree on the network graph.

The following therefore holds:

1. The optimum number of transmissions require per datum (data received at the sink from all sources) for this protocol is equal to the number of edges in the minimum Steiner tree in the network which contains nodes set ($S_{1\ldots\ldots}$Sn, R).

Corollary: Assuming an arbitrary placement of sources and general network graph G, the task of doing data centric routing with optimal data aggregation is NP-hard. This follows from the NP completeness of the minimum Steiner problem on graphs. (Bhaskar et al, 2010)(Huang et al, 2009)

Among the many data aggregation schemes Shortest path tree, Center at Nearest Source and Greedy incremental, I chose the Greedy incremental in which the aggregation tree is built sequentially .at first the tree only consists of the shortest path between nearest source and sink. At each step after that, the next closest is connected to the tree.

3.5 Performance Parameters

Energy Saving: aggregating information coming from sources reduces the number of transmissions, which in turn saves energy.

Delay: There is latency associated with data aggregation as data from nearer sources is held at aggregators waiting for data from far sources in order to combine them.

Robustness: Since energy is saved there is a decrease in marginal energy cost of connecting additional sources to the sink. This provides some degree of robustness to the sensed phenomena.

Factor which affect aggregation are the position or placement of the source nodes, the number of the nodes and the topology of the communication network. We look at two possible source placement models ie.Random source model and our optimal placement model.

Random Source Model

The sources are placed at random, not necessarily clustered near each other.

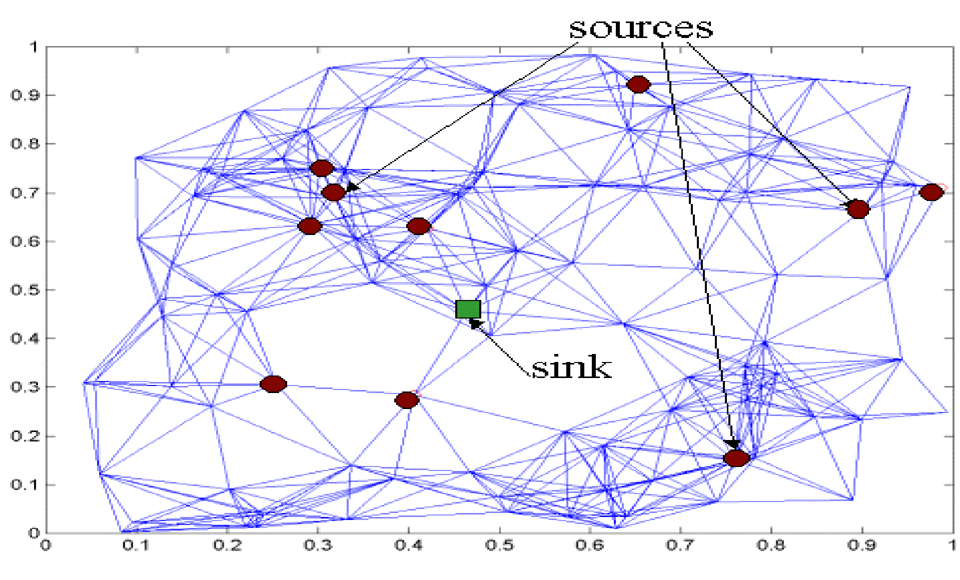


Fig 3.1.3 (Bhaskar et al, 2010) Random Source Model

Optimal Source Model

The sources are placed deterministically according to their communication radius and sensing radius covering the whole field there by achieving optimum coverage.

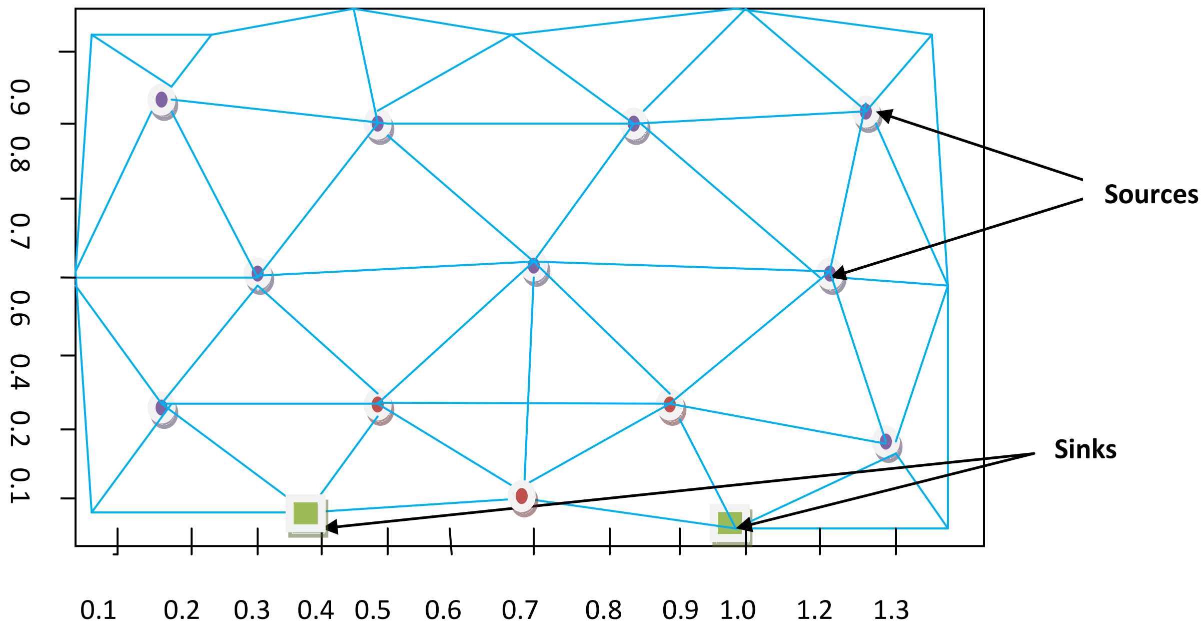


Fig 3.1.4 optimal source model

Theoretical Analysis

We analyze the analytical bounds of the energy cost and savings brought by data aggregation. The greatest gains are obtained when the sources are close together and far away from the sink.

Let d_i be the shortest distance from the source S_i to the sink in the graph. As per datum the total transmissions needed for Random source model N_R is:

$$N_R = d_1 + d_2 + d_3 + \cdots d_n = sum(d_i) \quad (5)$$

Let the number of transmissions needed for optimal source model be N_0

Then $N_0 \leq N_R$ must hold for it to be better.

Proof:

Doing data aggregation optimally decreases the minimum number of edges needed compared to when the sources send information only using the shortest path.

Definition: let X be the diameter of a set S of nodes in a graph G.

If the source nodes $S_1, S_2.....S_n$ have a diameter $X \geq 1$ the total number of transmissions N_o required for optimal data aggregation satisfies the following bounds:

$N_0 \leq (n-1)X + \min(d_i)$ (6)

$N_0 \geq (n-1) + \min(d_i)$ (7)

Proof: (2) by constructing a data aggregation tree which consists of $(n-1)$ sources sending packets to the remaining source which is nearest to the sink .This tree has no more than

$N_0 \leq (n-1)X + \min(d_i)$ Edges hence optimum tree must have no more than this.

Definition: Fractional energy saving (FS) in Optimal Source model

$FS = (N_R - N_0)/(N_R)$ $\quad 0 \leq FS \leq 1$ (8)

The upper and lower bounds of FS derived from (6) and (7) are:

$FS \geq 1 - ((n-1)X + \min(d_i))/sum(d_i)$ (9)

$FS \leq 1 - (\min(d_i) + (n-1))/sum(d_i)$ (10)

Assume that all the sources are at the same shortest –path distance from the sink that is $\min(d_i) = \max(d_i) = d$

Then we have:

$1 - \frac{((n-1)X+d)}{nd} \leq \text{FS} \leq 1 - \frac{(d+n-1)}{nd}$ (11)

$\text{Lim}_{d \to \infty} \text{FS} = 1 - 1/n$

Data Acquisition

- Using OMNeT 4.2.1 simulator I will run simulation on Data aggregation and node placement. I will test efficiency based on the time delays of data to reach the sink node.
- Graphs will be plotted to see relative improvements. Sensor data can be retrieved from the archive or directly from the base station and used by the control system to calculate the irrigation time.

Data querying

- May be time driven using time intervals

Challenges

- Setting up the sensors for communication between one another might be challenging as it is sometimes difficult to get a relatively clear line of site between nodes.
- Wireless spectrum is usually in the unlicensed low power range that cannot penetrate hills or dense vegetation. Common frequencies include 900MHz, 2.4 GHz and 5.8 GHZ. Nodes and antenna placement must be done in a way that prevents crops from absorbing signals.
- Internet connection availability

CHAPTER 4

EXPERIMENTAL RESULTS

Using wireless sensor simulator and OMNeT++ 4.21 simulator we investigated the effect of node placement on the performance of the network in terms of delays in packets delivery from source to sink against increase in network size. We also investigated on the effect of data aggregation on the network's efficiency and lifetime which we believe is of paramount importance in any design of wireless sensor networks. We go on to discuss our findings.

Assumption: Network connectivity between nodes is reliable.

Node Placement

Hops	**Topology 1**	**Topology 2**	**Topology 3**	**Topology 4**	**Ring Topology**	**Star Topology**	**Kite Topology**	**Linear Topology**	**Pentagonal Topology**	**Mesh Topology**
1	0.000576005	0.000556506	0.000453506	0.001586131	0.000576055	0.000456506	0.000473506	0.001086131	0.000676055	0.003586131
2	0.005495047	0.006823967	0.005824977	0.005496049	0.005495247	0.006623967	0.005994977	0.001296049	0.006495247	0.005696049
3	0.00887791	0.013357656	0.014397691	0.007987913	0.008187991	0.013257656	0.016397969	0.001587913	0.008088991	0.008987913
4	0.009849369	0.015329267	0.015929369	0.010949334	0.009849569	0.013329267	0.017939369	0.001949334	0.009859569	0.011949334
5	0.01054678	0.017856824	0.016898204	0.010946781	0.01074878	0.015856824	0.018298204	0.002094678	0.010096878	0.013094678
6	0.011044667	0.017896278	0.016898206	0.012548671	0.01124867	0.015896278	0.018398239	0.005254867	0.011604867	0.015054667
7	0.012445768	0.017846209	0.016899212	0.013545868	0.012745768	0.017988809	0.018450981	0.007954587	0.012745768	0.017545768
8	0.012345686	0.017896256	0.016898313	0.013945686	0.012745686	0.017999956	0.018648798	0.010945686	0.013396569	0.019545686
9	0.012547721	0.017896451	0.016898276	0.013847722	0.014547721	0.018619645	0.018499276	0.01184772	0.014054772	0.020254772
10	0.012946335	0.017896281	0.016898274	0.014948935	0.014946335	0.018896281	0.018599292	0.013948935	0.014056335	0.021294634

Table 4.1 Comparison of Topologies

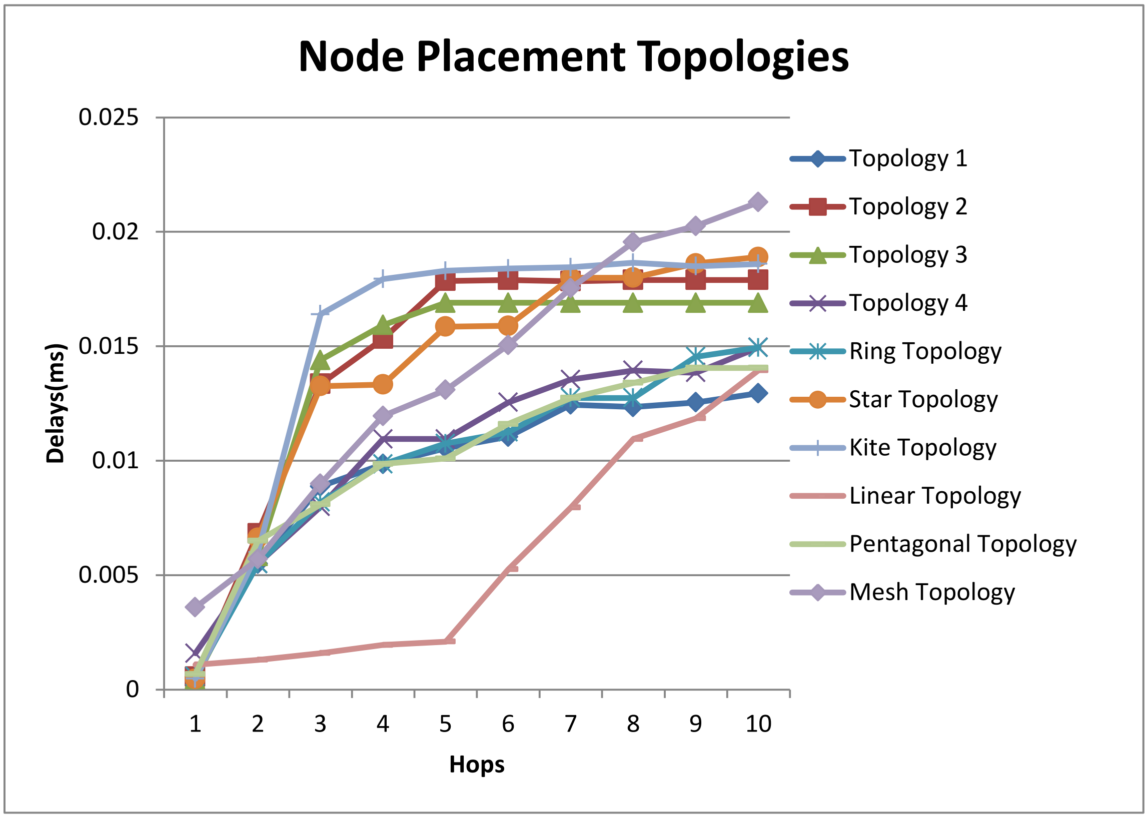


Fig 4.1 Topologies Graph

Topology1 performance: 2.5-8.4 ev/s.

Topology 2 performance: 2.5-6.1 ev/s.

Topology 3 performance: 2.5-6.8 ev/s

Topology 4 performance: 2.5-7.1 ev/s.

Ring Topology performance: 2.5-7.8 ev/s.

Star Topology performance: 2.5-8.1 ev/s.

Kite Topology performance: 2.5-6.5 ev/s

Linear Topology performance: 2.5-6.1 ev/s.

Pentagonal Topology performance: 2.5-7.6 ev/s.

Mesh Topology performance: 2.5-9.1 ev/s.

Discussion of Findings

Analyzing Fig 4.1 we can see that linear topology is very good for short distances between sources and sink that is the numbers of hops are few so it will be ideal for small areas of less than 10 sensors per 1000 m^2.

Pentagonal and Topology 1 are almost the same in delays but pentagonal gets poorer as the network size increases .The other problem with pentagonal is field coverage which not better than topology 1.

Topology 2 and 3 increases delays up to 5 hops and then increase in network size becomes insignificant on delays for up to 10 hops then it gradually increases again. We can also observe that they are less efficient than Topology 1, topology 4, Ring and Pentagonal topologies.

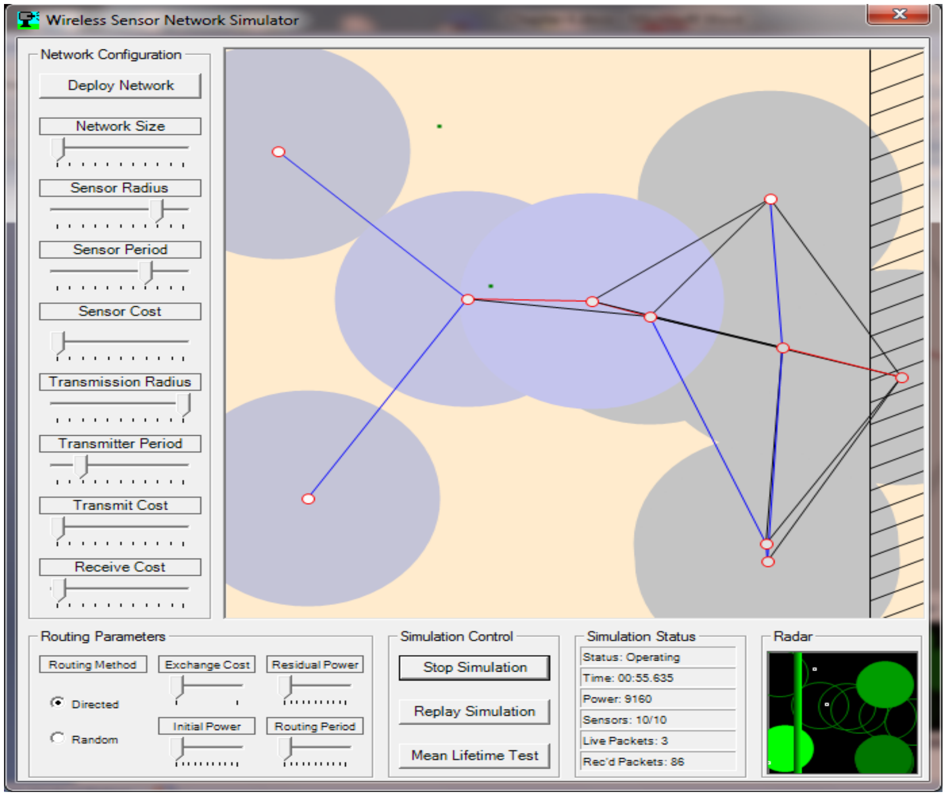
We see that for less than 10 hops linear topology (Bus) is ideal if we want to sacrifice coverage to network speed but topology 1 is better as it has both good network performance and good coverage.

Pentagonal and Ring are also good but they also have one drawback of coverage but we desire both network efficiency in terms of packets or data forwarding and sensing area coverage. This makes Topology 1 ideal.

Mesh shows a gradual increase in delays as network size increases as the routing table also increases and becomes complex due to many routes to be considered for packet forwarding and also it will need more energy .Mesh is advantageous if we are only concerned with coverage ,and non battery powered wireless networks .

The delays for all the topologies increases as the network size increases which means as the network increases there are many bottle necks which come into play, it also shows that the topology itself influence network performance and behavior.

Optimal Model using Data aggregation



Network life time Test

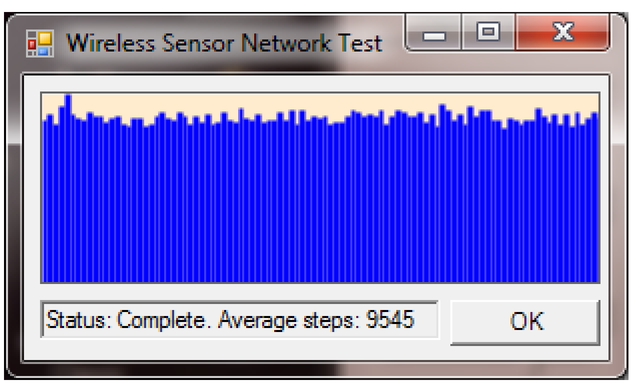


Fig 4.2

Optimal Model without Data Aggregation

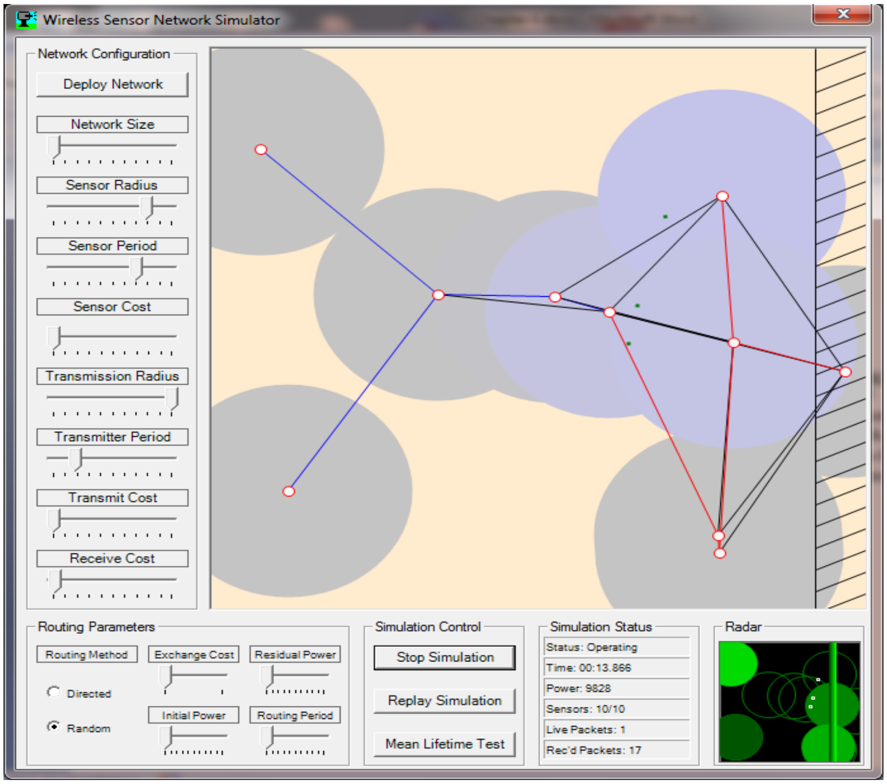


Fig 4.3

Control Model without Aggregation

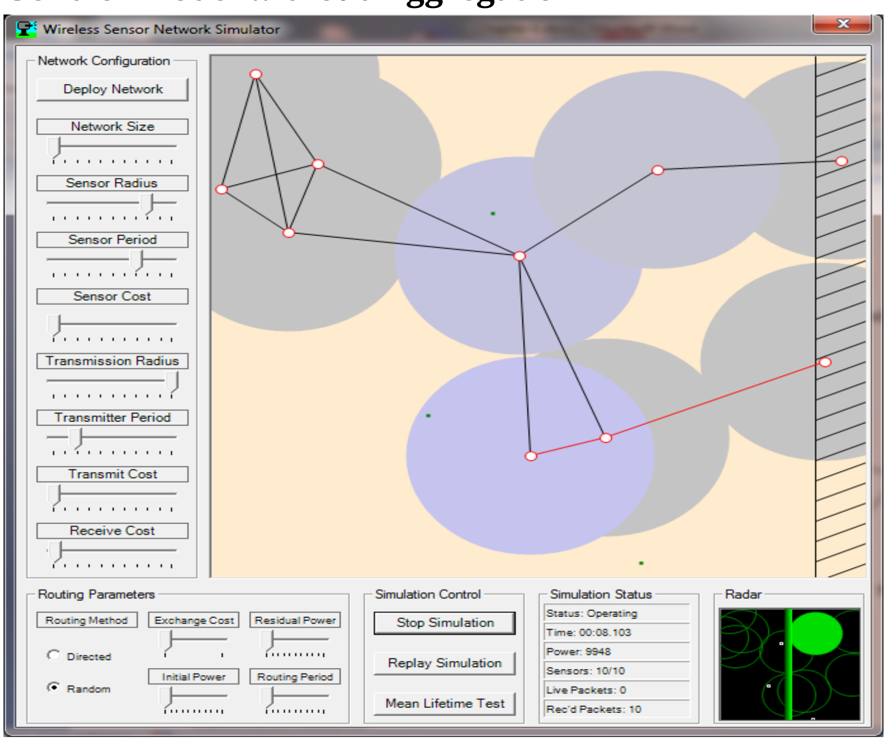


Fig 4.4

Control Model with Aggregation

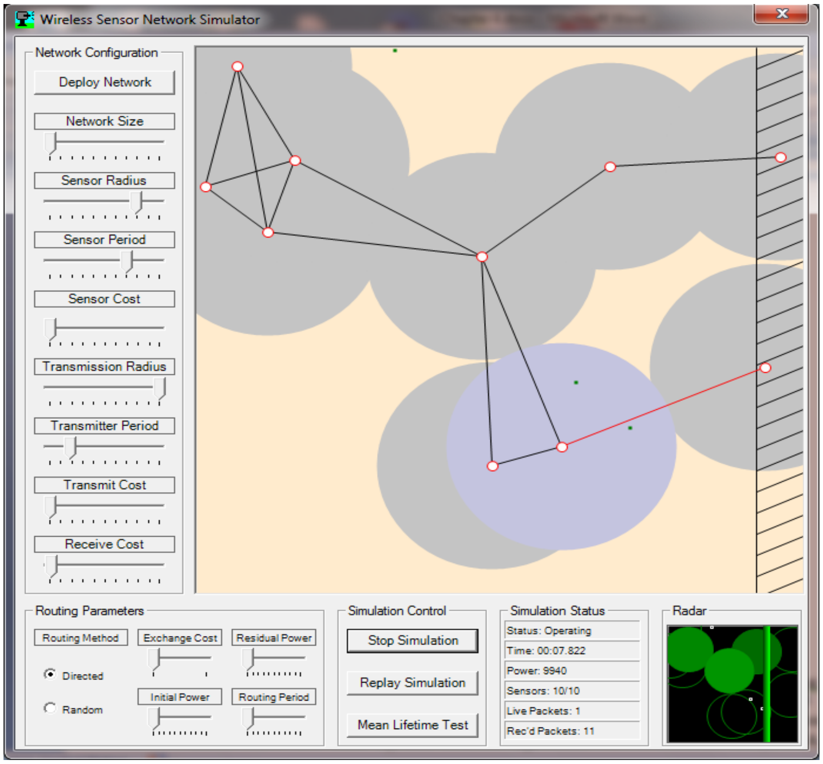


Fig 4.5

Data collection

Optimal topology **with aggregation**

time	power	sensor	received	pkts	efficiency(pkts/s)	Power utilization(j/pkt)
0.15	269	10	25	25	166.6666667	10.76
0.3	569	10	51	26	86.66666667	21.88461538
0.45	367	10	76	25	55.55555556	14.68
1	473	10	102	26	26	18.19230769
1.15	222	9	129	27	23.47826087	8.222222222
1.3	163	9	154	25	19.23076923	6.52
1.45	104	9	180	26	17.93103448	4
2	245	8	206	26	13	9.423076923
2.15	286	7	233	27	12.55813953	10.59259259
2.3	120	7	258	25	10.86956522	4.8
					43.19566582	**10.90748148**

Table 4.2

Optimal Topology Without Aggregation

time	power	sensor	received	pkts	efficiency(pkts/s)	Power utilization(j/pkt)
0.15	170	10	24	24	160	7.083333333
0.3	184	10	49	25	83.33333333	7.36
0.45	348	10	75	26	57.77777778	13.38461538
1	273	9	101	26	26	10.5
1.15	256	7	127	26	22.60869565	9.846153846
1.3	342	7	152	25	19.23076923	13.68
1.45	435	6	179	27	18.62068966	16.11111111
2	312	6	204	25	12.5	12.48
2.15	242	6	231	27	12.55813953	8.962962963
2.3	339	5	255	24	10.43478261	14.125
					42.30641878	**11.35331766**

Table4.3

Control Topology With Aggregation

time	power	sensor	received	packets	efficiency(pkts/s)	Power utilization(j/pkts)
0.15	58	10	23	23	153.3333333	2.52173913
0.3	114	10	48	25	83.33333333	4.56
0.45	475	9	73	25	55.55555556	19
1	149	9	100	27	27	5.518518519
1.15	143	9	126	26	22.60869565	5.5
1.3	123	8	150	24	18.46153846	5.125
1.45	98	8	177	27	18.62068966	3.62962963
2	79	7	204	27	13.5	2.925925926
2.15	212	6	235	31	14.41860465	6.838709677
2.3	295	5	255	20	8.695652174	8.428571429
					41.55274028	**6.404809431**

Table 4.4

Control topology without aggregation

time	power	sensor	received	packets	efficiency(pkts/s)	Power utilization(j/pkt)
0.15	65	10	23	23	153.3333333	2.826086957
0.3	112	10	48	25	83.33333333	4.48
0.45	467	9	74	26	57.77777778	17.96153846
1	149	9	99	25	25	5.96
1.15	154	9	124	25	21.73913043	6.16
1.3	123	8	149	25	19.23076923	4.92
1.45	102	7	174	25	17.24137931	4.08
2	80	7	200	26	13	3.076923077
2.15	222	5	226	26	12.09302326	8.538461538
2.3	312	4	250	24	10.43478261	13
					41.31835293	**7.100301003**

Table 4.5

Efficiency

Time	Control topology with aggregation	Control Topology without aggregation	Optimal topology with aggregation	Optimal topology without aggregation
0.15	153.3333333	153.3333333	166.6666667	160
0.3	83.33333333	83.33333333	86.66666667	83.33333333
0.45	55.55555556	57.77777778	55.55555556	57.77777778
1	27	25	26	26
1.15	22.60869565	21.73913043	23.47826087	22.60869565
1.3	18.46153846	19.23076923	19.23076923	19.23076923
1.45	18.62068966	17.24137931	17.93103448	18.62068966
2	13.5	13	13	12.5
2.15	14.41860465	12.09302326	12.55813953	12.55813953
2.3	8.695652174	10.43478261	10.86956522	10.43478261

Table4.6

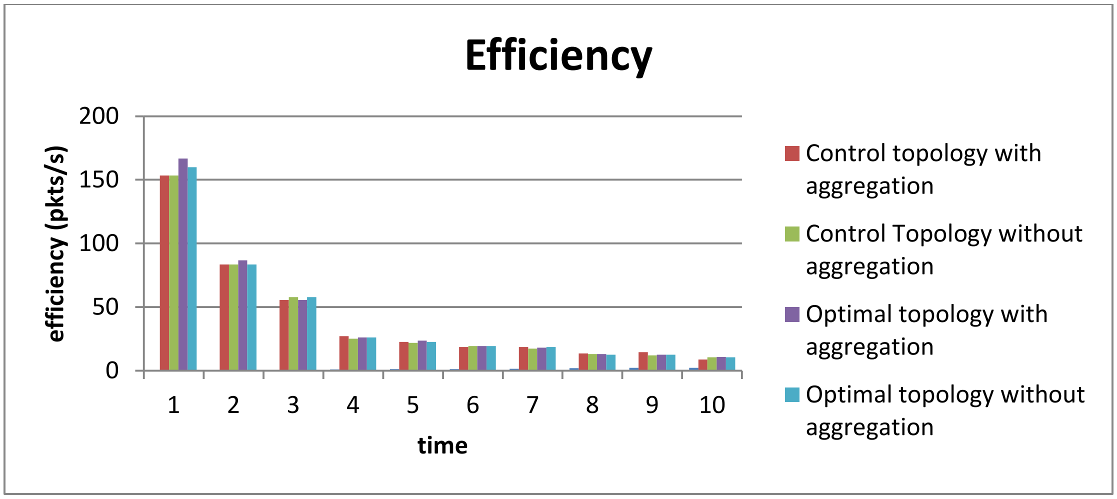


Fig 4.6

Discussion of findings

We can see that for both topologies the one with data aggregation has better efficiency than without .we also observe that our design has more efficiency than the control. This comes to the conclusion that data aggregation is ideal for improved network performance and resource conservation in WSN.

Power Utilization

time	Optimal with aggregation	Topology with aggregation	optimal without aggregation	Topology without aggregation
0.15	2.52173913	10.76	2.826086957	7.083333333
0.3	4.56	21.88461538	4.48	7.36
0.45	19	14.68	17.96153846	13.38461538
1	5.518518519	18.19230769	5.96	10.5
1.15	5.5	8.222222222	6.16	9.846153846
1.3	5.125	6.52	4.92	13.68
1.45	3.62962963	4	4.08	16.11111111
2	2.925925926	9.423076923	3.076923077	12.48
2.15	6.838709677	10.59259259	8.538461538	8.962962963
2.3	8.428571429	4.8	13	14.125

Table 4.7

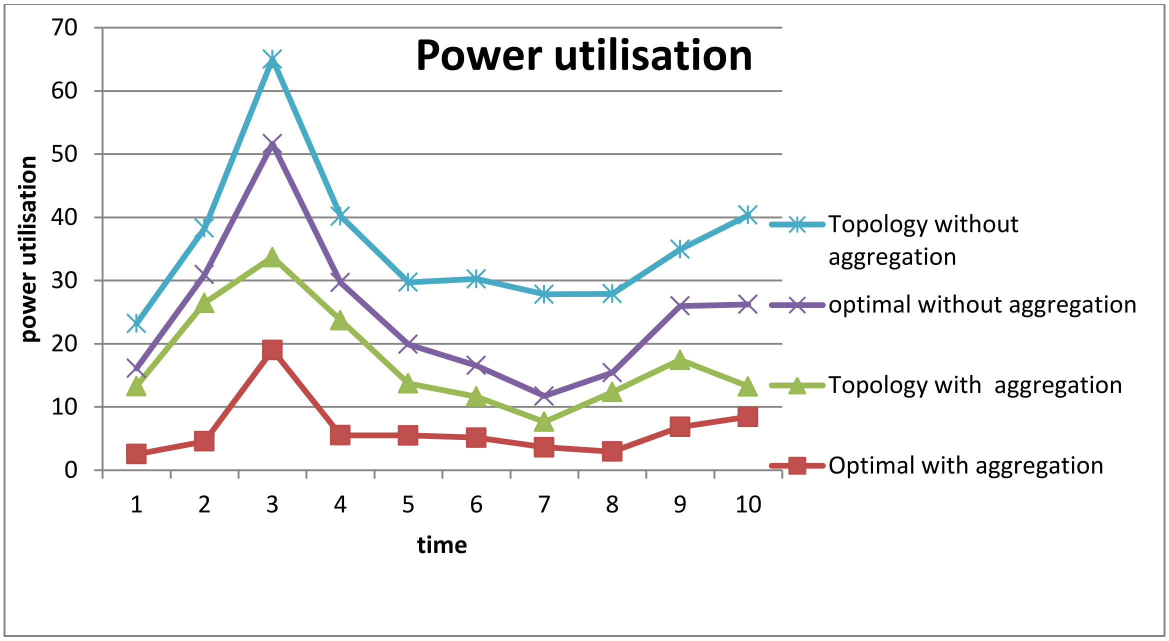


Fig 4.7

Discussion of findings

We can observe that both topologies with data aggregation have lower power utilization than without. This is due to the fact that data aggregation reduces network load by eliminating redundancy which degrades the performance of the network by increasing collisions, delay, and energy consumption thereby minimizing transmission power (Monaco et al, 2006).In network processing of data consumes less energy than data transmission so we can capitalize on that.

Network life

Net (nodes/km2)	**10**	**25**	**50**	**75**	**100**	**125**	**150**	**175**
deaths	0	1	2	4	11	15	30	47
power consumption	5947	7791	11822	21244	29039	36770	147101	1770975

Table 4.8

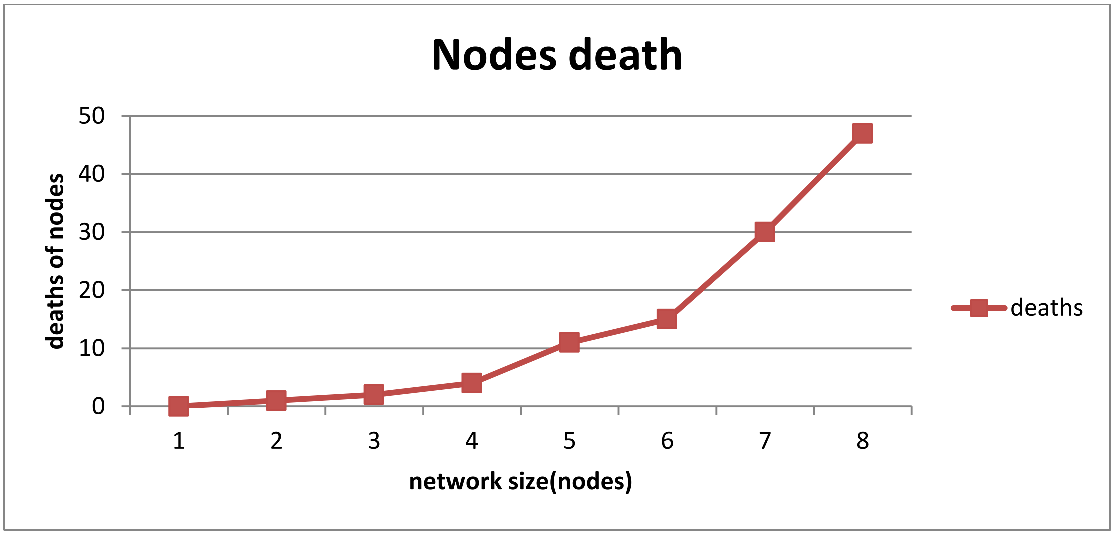


Fig 4.8

Discussion of findings

We also investigated under simulation the rate of death of nodes due to power dissipation and we can observe that as less death for up to 100 nodes per square km but for network sizes greater than that we see a drastic decrease in network lifetime due to increased node mortality. As number of nodes increases connections also between nodes increases more routing and throughput requires more energy and transmission distance also increases thereby consuming more energy so the nodes will lose energy at faster rates. (Venkata et al, 2010)

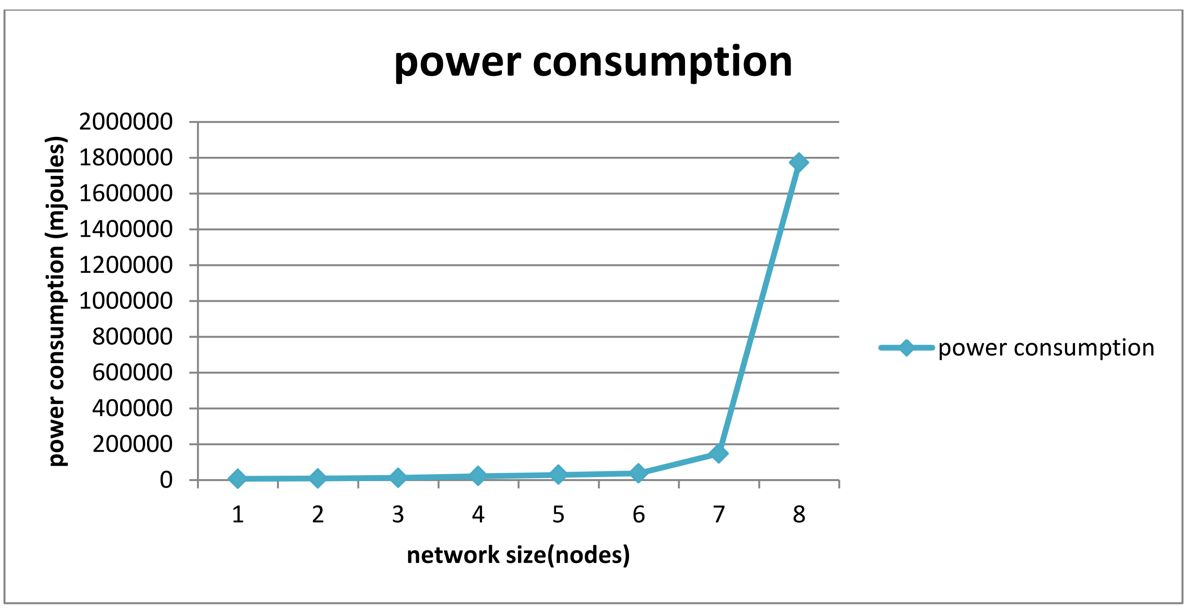


Fig 4.9

Discussion of findings

Power consumption increases insignificantly up to 125 nodes per km^2 but above that there is a sharp increase which is constituted by increased interference and in network processing between nodes due to their density. (Monaco et al, 2006)

CHAPTER 5

CONCLUSIONS

Wireless Sensor Networks are mainly deployed for monitoring purposes in various fields. As many sensors detect the same event and try to forward the data to other nodes, data is redundant and degrades the performance of the network by increasing collisions, delay, and energy consumption. Data Aggregation techniques are used in some applications to reduce the redundancy in forwarded packets. In these techniques, packets are aggregated at intermediate nodes and the correlated data is forwarded from one node to another,(Monaco et al, 2006). Also, as sensor nodes are energy constrained, energy efficiency is one of the primary concerns in finding suitable protocols for these networks.

To enhance the packet-level reliability and reduce energy consumption, we developed a reliable network topology which incorporates data aggregation using directed diffusion and duplicate suppression techniques. For WSN, many protocols have been proposed that provide reliability and good transmission ranges with low power consumption and we found ZigBee protocol being the best technology to date. It has many advantages which includes its portability, long range transmission (up to 1 km), free frequency bands and scalability and low prices. It is based on IEEE802.15.4 MAC and PHY and have data rate up to 250kbps and provides 16 channels in the unlicensed 2.4GHz band. It is supported with JN5148 wireless microcontroller and modules. (NA, 2010)

Our simulation results show that node placement and data aggregation techniques improve energy efficiency and the packet forwarding even in large in highly dense WSN. However, latency tends to increase under congested scenarios because of by increasing collisions, delay, and energy consumption. In general, an increase in latency would affect the performance of the network.

In the future, we would like to extend our research into real time implementation of these Topologies and data aggregation techniques.

KEY WORDS

Wireless sensor network, Node placement, Data aggregation, Precision irrigation.

LIST OF FIGURES

LIST OF TABLES

ANNEXES

A Packet Definition

```
// Disable warnings about unused variables, empty switch stmts, and so on:
#ifdef _MSC_VER
#  pragma warning(disable:4101)
#  pragma warning(disable:4065)
#endif

#include <iostream>
#include <sstream>
#include "Packet_m.h"

// Template rule which fires if a struct or class doesn't have operator<<
template<typename T>
std::ostream& operator<<(std::ostream& out,const T&) {return out;}

// Another default rule (prevents compiler from choosing base class' doPacking())
template<typename T>
void doPacking(cCommBuffer *, T& t) {
   throw cRuntimeError("Parsim error: no doPacking() function for type %s or its base class (check .msg and
_m.cc/h files!)",opp_typename(typeid(t)));
}

template<typename T>
void doUnpacking(cCommBuffer *, T& t) {
   throw cRuntimeError("Parsim error: no doUnpacking() function for type %s or its base class (check .msg and
_m.cc/h files!)",opp_typename(typeid(t)));
}




Register_Class(Packet);

Packet::Packet(const char *name, int kind) : cPacket(name,kind)
{
   this->srcAddr_var = 0;
   this->destAddr_var = 0;
   this->hopCount_var = 0;
}

Packet::Packet(const Packet& other) : cPacket(other)
{
   copy(other);
}

Packet::~Packet()
{
}

Packet& Packet::operator=(const Packet& other)
{
   if (this==&other) return *this;
   cPacket::operator=(other);
   copy(other);
   return *this;
}
```

```
void Packet::copy(const Packet& other)
{
   this->srcAddr_var = other.srcAddr_var;
   this->destAddr_var = other.destAddr_var;
   this->hopCount_var = other.hopCount_var;
}

void Packet::parsimPack(cCommBuffer *b)
{
   cPacket::parsimPack(b);
   doPacking(b,this->srcAddr_var);
   doPacking(b,this->destAddr_var);
   doPacking(b,this->hopCount_var);
}

void Packet::parsimUnpack(cCommBuffer *b)
{
   cPacket::parsimUnpack(b);
   doUnpacking(b,this->srcAddr_var);
   doUnpacking(b,this->destAddr_var);
   doUnpacking(b,this->hopCount_var);
}

int Packet::getSrcAddr() const
{
   return srcAddr_var;
}

void Packet::setSrcAddr(int srcAddr)
{
   this->srcAddr_var = srcAddr;
}

int Packet::getDestAddr() const
{
   return destAddr_var;
}

void Packet::setDestAddr(int destAddr)
{
   this->destAddr_var = destAddr;
}

int Packet::getHopCount() const
{
   return hopCount_var;
}

void Packet::setHopCount(int hopCount)
{
   this->hopCount_var = hopCount;
}

class PacketDescriptor : public cClassDescriptor
{
  public:
   PacketDescriptor();
```

```
    virtual ~PacketDescriptor();

    virtual bool doesSupport(cObject *obj) const;
    virtual const char *getProperty(const char *propertyname) const;
    virtual int getFieldCount(void *object) const;
    virtual const char *getFieldName(void *object, int field) const;
    virtual int findField(void *object, const char *fieldName) const;
    virtual unsigned int getFieldTypeFlags(void *object, int field) const;
    virtual const char *getFieldTypeString(void *object, int field) const;
    virtual const char *getFieldProperty(void *object, int field, const char *propertyname) const;
    virtual int getArraySize(void *object, int field) const;

    virtual std::string getFieldAsString(void *object, int field, int i) const;
    virtual bool setFieldAsString(void *object, int field, int i, const char *value) const;

    virtual const char *getFieldStructName(void *object, int field) const;
    virtual void *getFieldStructPointer(void *object, int field, int i) const;
};

Register_ClassDescriptor(PacketDescriptor);

PacketDescriptor::PacketDescriptor() : cClassDescriptor("Packet", "cPacket")
{
}

PacketDescriptor::~PacketDescriptor()
{
}

bool PacketDescriptor::doesSupport(cObject *obj) const
{
    return dynamic_cast<Packet *>(obj)!=NULL;
}

const char *PacketDescriptor::getProperty(const char *propertyname) const
{
    cClassDescriptor *basedesc = getBaseClassDescriptor();
    return basedesc ? basedesc->getProperty(propertyname) : NULL;
}

int PacketDescriptor::getFieldCount(void *object) const
{
    cClassDescriptor *basedesc = getBaseClassDescriptor();
    return basedesc ? 3+basedesc->getFieldCount(object) : 3;
}

unsigned int PacketDescriptor::getFieldTypeFlags(void *object, int field) const
{
    cClassDescriptor *basedesc = getBaseClassDescriptor();
    if (basedesc) {
        if (field < basedesc->getFieldCount(object))
            return basedesc->getFieldTypeFlags(object, field);
        field -= basedesc->getFieldCount(object);
    }
    static unsigned int fieldTypeFlags[] = {
        FD_ISEDITABLE,
        FD_ISEDITABLE,
        FD_ISEDITABLE,
```

```
    };
    return (field>=0 && field<3) ? fieldTypeFlags[field] : 0;
}

const char *PacketDescriptor::getFieldName(void *object, int field) const
{
    cClassDescriptor *basedesc = getBaseClassDescriptor();
    if (basedesc) {
        if (field < basedesc->getFieldCount(object))
            return basedesc->getFieldName(object, field);
        field -= basedesc->getFieldCount(object);
    }
    static const char *fieldNames[] = {
        "srcAddr",
        "destAddr",
        "hopCount",
    };
    return (field>=0 && field<3) ? fieldNames[field] : NULL;
}

int PacketDescriptor::findField(void *object, const char *fieldName) const
{
    cClassDescriptor *basedesc = getBaseClassDescriptor();
    int base = basedesc ? basedesc->getFieldCount(object) : 0;
    if (fieldName[0]=='s' && strcmp(fieldName, "srcAddr")==0) return base+0;
    if (fieldName[0]=='d' && strcmp(fieldName, "destAddr")==0) return base+1;
    if (fieldName[0]=='h' && strcmp(fieldName, "hopCount")==0) return base+2;
    return basedesc ? basedesc->findField(object, fieldName) : -1;
}

const char *PacketDescriptor::getFieldTypeString(void *object, int field) const
{
    cClassDescriptor *basedesc = getBaseClassDescriptor();
    if (basedesc) {
        if (field < basedesc->getFieldCount(object))
            return basedesc->getFieldTypeString(object, field);
        field -= basedesc->getFieldCount(object);
    }
    static const char *fieldTypeStrings[] = {
        "int",
        "int",
        "int",
    };
    return (field>=0 && field<3) ? fieldTypeStrings[field] : NULL;
}

const char *PacketDescriptor::getFieldProperty(void *object, int field, const char *propertyname) const
{
    cClassDescriptor *basedesc = getBaseClassDescriptor();
    if (basedesc) {
        if (field < basedesc->getFieldCount(object))
            return basedesc->getFieldProperty(object, field, propertyname);
        field -= basedesc->getFieldCount(object);
    }
    switch (field) {
        default: return NULL;
    }
}
```

```
int PacketDescriptor::getArraySize(void *object, int field) const
{
   cClassDescriptor *basedesc = getBaseClassDescriptor();
   if (basedesc) {
      if (field < basedesc->getFieldCount(object))
         return basedesc->getArraySize(object, field);
      field -= basedesc->getFieldCount(object);
   }
   Packet *pp = (Packet *)object; (void)pp;
   switch (field) {
      default: return 0;
   }
}

std::string PacketDescriptor::getFieldAsString(void *object, int field, int i) const
{
   cClassDescriptor *basedesc = getBaseClassDescriptor();
   if (basedesc) {
      if (field < basedesc->getFieldCount(object))
         return basedesc->getFieldAsString(object,field,i);
      field -= basedesc->getFieldCount(object);
   }
   Packet *pp = (Packet *)object; (void)pp;
   switch (field) {
      case 0: return long2string(pp->getSrcAddr());
      case 1: return long2string(pp->getDestAddr());
      case 2: return long2string(pp->getHopCount());
      default: return "";
   }
}

bool PacketDescriptor::setFieldAsString(void *object, int field, int i, const char *value) const
{
   cClassDescriptor *basedesc = getBaseClassDescriptor();
   if (basedesc) {
      if (field < basedesc->getFieldCount(object))
         return basedesc->setFieldAsString(object,field,i,value);
      field -= basedesc->getFieldCount(object);
   }
   Packet *pp = (Packet *)object; (void)pp;
   switch (field) {
      case 0: pp->setSrcAddr(string2long(value)); return true;
      case 1: pp->setDestAddr(string2long(value)); return true;
      case 2: pp->setHopCount(string2long(value)); return true;
      default: return false;
   }
}

const char *PacketDescriptor::getFieldStructName(void *object, int field) const
{
   cClassDescriptor *basedesc = getBaseClassDescriptor();
   if (basedesc) {
      if (field < basedesc->getFieldCount(object))
         return basedesc->getFieldStructName(object, field);
      field -= basedesc->getFieldCount(object);
   }
   static const char *fieldStructNames[] = {
```

```
        NULL,
        NULL,
        NULL,
    };
    return (field>=0 && field<3) ? fieldStructNames[field] : NULL;
}

void *PacketDescriptor::getFieldStructPointer(void *object, int field, int i) const
{
    cClassDescriptor *basedesc = getBaseClassDescriptor();
    if (basedesc) {
        if (field < basedesc->getFieldCount(object))
            return basedesc->getFieldStructPointer(object, field, i);
        field -= basedesc->getFieldCount(object);
    }
    Packet *pp = (Packet *)object; (void)pp;
    switch (field) {
        default: return NULL;
    }
}
```

B Routing (Aggregation)

```
#ifdef _MSC_VER
#pragma warning(disable:4786)
#endif

#include <map>
#include <omnetpp.h>
#include "Packet_m.h"


/**
 * Demonstrates static routing, utilizing the cTopology class.
 */
class Routing : public cSimpleModule
{
  private:
    int myAddress;

    typedef std::map<int,int> RoutingTable; // destaddr -> gateindex
    RoutingTable rtable;

    simsignal_t dropSignal;
    simsignal_t outputIfSignal;

  protected:
    virtual void initialize();
    virtual void handleMessage(cMessage *msg);
};

Define_Module(Routing);


void Routing::initialize()
{
    myAddress = getParentModule()->par("address");
```

```
    dropSignal = registerSignal("drop");
    outputIfSignal = registerSignal("outputIf");

    //
    // Brute force approach -- every node does topology discovery on its own,
    // and finds routes to all other nodes independently, at the beginning
    // of the simulation. This could be improved: (1) central routing database,
    // (2) on-demand route calculation
    //
    cTopology *topo = new cTopology("topo");

    std::vector<std::string> nedTypes;
    nedTypes.push_back(getParentModule()->getNedTypeName());
    topo->extractByNedTypeName(nedTypes);
    EV << "cTopology found " << topo->getNumNodes() << " nodes\n";

    cTopology::Node *thisNode = topo->getNodeFor(getParentModule());

    // find and store next hops
    for (int i=0; i<topo->getNumNodes(); i++)
    {
        if (topo->getNode(i)==thisNode) continue; // skip ourselves
        topo->calculateUnweightedSingleShortestPathsTo(topo->getNode(i));

        if (thisNode->getNumPaths()==0) continue; // not connected

        cGate *parentModuleGate = thisNode->getPath(0)->getLocalGate();
        int gateIndex = parentModuleGate->getIndex();
        int address = topo->getNode(i)->getModule()->par("address");
        rtable[address] = gateIndex;
        EV << "  towards address " << address << " gateIndex is " << gateIndex << endl;
    }
    delete topo;
}

void Routing::handleMessage(cMessage *msg)
{
    Packet *pk = check_and_cast<Packet *>(msg);
    int destAddr = pk->getDestAddr();

    if (destAddr == myAddress)
    {
        EV << "local delivery of packet " << pk->getName() << endl;
        send(pk, "localOut");
        emit(outputIfSignal, -1); // -1: local
        return;
    }

    RoutingTable::iterator it = rtable.find(destAddr);
    if (it==rtable.end())
    {
        EV << "address " << destAddr << " unreachable, discarding packet " << pk->getName() << endl;
        emit(dropSignal, (long)pk->getByteLength());
        delete pk;
        return;
    }

    int outGateIndex = (*it).second;
```

```
   EV << "forwarding packet " << pk->getName() << " on gate index " << outGateIndex << endl;
   pk->setHopCount(pk->getHopCount()+1);
   emit(outputIfSignal, outGateIndex);

   send(pk, "out", outGateIndex);
}
```

C Duplicate suppression Algorithm

- T_1 get data from neighboring nodes(labeled)
- T_2 Compare data
- If (duplicate)
- Take one and discard the rest
- T_3 calculate the average of this data
- T_4 Forward the averaged data to the next sink on request or at time intervals

RÉFÉRENCES

[1] Reberto, Localization in Mobile Wireless Sensor Networks,World Academy of Science, Engineering and Technology 63 2010,pp 25-32

[2] http://www.omega.com/prodinfo/wirelesssensors.html, (Accesed January 4th 2012).

[3] Andrew G.Ristevey, John D.Lea-Cox, Smarter irrigation management wireless sensor systems, RistveyMD08, (2006), pp 11.

[4] K.Ramanan and E.Baburaj, data gathering algorithms for wireless sensor networks: a Survey, international journal of ad hoc, sensor & ubiquitous computing (ijasuc) vol.1, no.4, December 2010doi: 10.5121/ijasuc.2010.1410, (2010), pp 102

[5] Adekeyede A. Elusoji, A. U. Rufai, Available at: http://www.academicleadership.org/article/an-overview-of-smart-sensor-technology, Accessed Mar 2 2012.pp 27-30

[6] R. Jordan, R. and Abdallah, C.A, "Wireless communications and networking: an overview," Report, Elect. And Comp. Eng. Dept, Univ. New Mexico, (2009), pp 21-26

[7] Chong, Chee-Yee and Srikanta P. Kumar, "Sensor Networks: Evolution, Opportunities, and Challenges", *Proceedings of the IEEE, 91*, 8, 1247-1256, (2008), pp 20-22

[8]Culler, D.D.Estrin and M.Srivastava, "Overview of Sensor Networks", *Computer*, Washington, DC IEEE Computer Society, (2007), pp 40-49.

[9] Luis Ruiz-Garcia, Loredana Lunadei, Pilar Barreiro and Jose Ignacio Robla, a Review of Wireless Sensor Technologies and Applications in Agriculture and Food Industry: State of the Art and Current Trends. www.mdpi.com/journal/sensors, ISSN 1424-8220, published June 2009, pp 6-9

[10] Marc Van Lersel and Stephanie Bumett, Using water sensors for efficient irrigation in greenhouse, 2010, pp 4-7

[11] Jason Lester Hill, System Architecture for Wireless Sensor Networks, 2003.7-8

[12] Kazem Sohraby, Daniel Minoli and Taieb znati, Wireless sensor networks Technology, protocols, and applications, Wiley Interscience Publication, (2007),pp7-10

[13] www.icir.org/models/simulators.html, accessed 15/10/2011.

[14] Yunseop Kim, William Iversen and Francis J. Pierce ,Instrumentation and Control for Wireless Sensor Network for Automated Irrigation , ASABE Annual International Meeting Sponsored by ASABE Portland Convention Center Portland, Oregon, (2006), pp 7-8

[15]Ignacio Solis and Kaita Obraczka .The impact of Timing in Data Aggregation for Sensor Networks, Computer Engineering Department University of California (ND), pp9-11

[16] Karim Seada, Energy Efficient Forwarding Strategies for Geographical Routing in Lossy Wireless Sensor Networks, (ND), pp 6-9

[17]J Balendonk, J Hemming, B A J Van Tuij, L Incrocci, A Pardossi, P Marzialetti, Sensors and Wireless Sensor Networks For Irrigation Management under Deficit Conditions (FLOW AID), 2009, pp 3-5

[18] Yunseop (James) Kim, *Member*, Robert G. Evans, and William M. Iversen, Remote Sensing and Control of an Irrigation System Using a Distributed Wireless Sensor Network, *IEEE*, July 2008,pp 6-10

[19] Tahar Boutraa, Abdullah Akhkha, Abdulkhaliq Alshuaibi, Ragheid Atta
, Evaluation of the effectiveness of an automated irrigation system using wheat crops, agriculture and biology journal of North America ISSN print: 2151-7517, 2011, pp10-16

[20] Hamid Barati, Ali Movaghar, Ali Barati, and Arash azizi mazreah.A Review of Coverage and Routing for Wireless Sensor Networks World Academy of Science, Engineering and Technology 37 2008,pp 13-14

[21] Yujin Lim, Hak-Man Kim and Sanggil Kang, a Design of Wireless Sensor Networks for a Power Quality Monitoring System, Published in *Sensors* 2010 journals, www.mdpi.com/journal/sensors. ISSN 1424-8220.7

[22] Luis Javier García Villalba , Ana Lucila Sandoval Orozco, Alicia Triviño Cabrera and Cláudia Jacy Barenco Abbas,Routing Protocols in Wireless Sensor Networks Published in *Sensors* 2009 journals ,www.mdpi.com/journal/sensors .ISSN 1424-822, 2009, pp 45-46

[23] Kazem sohraby, Daniel minoli, Taieb znati, Wireless sensor networks technology, protocols, and applications.pdf published by John Wiley & Sons, Inc., Hoboken, New Jersey 2007, pp 53-60

[24] Wei Han, Three-tier wireless sensor network infrastructure for Environmental monitoring Kansas State University Manhattan, Kansas 2011, pp 70-71.

[25] Abhinav Valada, David Kohanbash, George Kantor, Design and Development of a Wireless Sensor Network System for Precision Agriculture Robotics Institute Carnegie Mellon University Pittsburgh Pennsylvania, June 2010, pp16-18.

[26] Amitabha Ghosh, Algorithmic Aspects of Throughput-Delay Performance For Fast Data Collection in Wireless Sensor Networks, Faculty of the USC Graduate School University Of Southern California, August 2010, pp4-5.

[27] FC Mwamuka Agriculture in Zimbabwe article, 1999.

[28] JM Makacho, The Effects of climate change in Zimbabwe Agriculture, 1999, pp2-4

[29]Gary Marks, Precision Irrigation: A method to save water and energy while increasing crop yield .A targeted approach for California Agriculture, March 2010, pp34-35

[30]FC Mwamuka, A Makwiro, The effects of climate change on Zimbabwe's Irrigation program, 1999, pp45-47
[31] Hilton Chikwiriro, Modeling a low cost data acquisition and irrigation sequencing system for a Greenhouse on an 8 bit PIC Microcontroller ,UZ,2010, pp 67-71.
[32] D.d.chaudhary, S.P Nayse, l.M.Waghmare Application of wireless sensor networks for greenhouse parameter control in Precision agriculture. International Journal of Wireless & Mobile Networks (IJWMN) Vol. 3, No. 1, February 2011, pp 56-58.

[33] Muhamad Azman Miskam, Azwan bin Nasirudin, Inzarulfaisham Abd. Rahim, Preliminary Design on the Development of Wireless Sensor Network for Paddy Rice Cropping Monitoring Application in Malaysia. http://www.eurojournals.com/ejsr.htm, Euro Journals Publishing, Inc. 2009, pp 23-25.
[34] Yunxia Chen, chen-nee Chua, and Qing Zhao, Sensor placement for maximizing lifetime per unit cost in wireless sensor networks. University of California, Davis, CA, 2009, pp 11-15
[35]Jamal N Al-Karaki, Raza Ul-Mustafa, Ahmed E-Kamal, Data aggregation and routing in wireless sensor networks: Optimal heuristic algorithms. IOWA state university, USA.2010, pp 56-60.
[36] Bhaskar Krishnamachari, Deborah Estrin, and Stephen Wicker, Modeling Data-Centric Routing in Wireless Sensor Networks, Department of Electrical Engineering Systems, University of Southern California.2010, pp 6-7.
[37] Tsung-Han Lin, Polly Huang, Sensor Data Aggregation for Resource Inventory Applications, National Taiwan University Taipei, Taiwan, 2009,pp 7-12.
[38] Sungrae CHO, Proactive Data Filtering Algorithm for Aggregation in Wireless Sensor Networks, IEICE TRANS. COMMUN., VOL.E91–B, NO.3 MARCH 2008,pp 13
[39] Vamsi Krishna Venkata Naga Nandanavanam, Energy-Efficient Reliable Sensor-To-Sink Data Transfer For Wireless Sensor Networks, Boise State University May 2010,pp 45-50
[40] U. Monaco, F. Cuomo, T. Melodia, F. Ricciato, M. Borghini, Understanding optimal data gathering in the energy and latency domains of a wireless sensor network, Computer Networks 50 (2006) 3564–3584, (2006).
[41] NA (no author), ZigBee_PRO_PB v1.2, NXP Laboratories UK Ltd 2010, pp 1-2.