Deployment and topology of a wireless sensor network for precision viticulture

Gregorio Corral-Torres[1], Carlos Garcia-Rubio

Departament of Telematic Engineering, Universidad Carlos III de Madrid, Spain

Abstract

Precision viticulture is a specialization of precision agriculture techniques applied to viticulture. Precision agriculture is the use of information system technologies applied to agricultural production. Some of the applicable technologies are; Wireless Sensor Networks (WSN), Global Positioning Systems (GPS), spectroscopy analysis of Near-Infrared (NIR), Geographic Information Systems (GIS). These systems provide means of observation, evaluation and control of agricultural activities. The farmers demand assistant systems to perform actions for saving time and avoiding risks. There are studies of maps crops and mesh-sampling techniques to predict the harvest volume in a vineyard with a certain varieties of grapes. The prediction is based on a previous study of crops over a period of three to four years. Along these three or four years a large volume of samples is taken to study several parameters. In this application area is where the wireless sensor networks technologies would have high incidence. In this context we intend to analyse, at first place, the specific characteristics of the operational environment of a vineyard. Second, we will analyse the most appropriate architecture for a sensor network in this environment. Application of wireless sensor networks technology can take many forms depending of environment, and implementation objectives. In this paper we discuss about the best procedure for deployment and the optimal topology of a wireless sensor network for viticulture.

Keywords: precision viticulture, WSN, ZigBee

1. Introduction

There are several productive sectors that have adopted and integrated different information technologies within their business models but these technologies have not a great impact yet in the viticulture sector. The sector have certain resistance to adopt these technologies, although agriculture sector do not have important technical restrictions for the deployment of sensor networks. Typically, these application environments allow use of some mechanism of external power supply, solar energy for example, and mobility is not usually a main requirement.

The barrier is that most farmers are not accustomed to use decision support systems (DSS). The farmers usually focus their efforts on production [1] and not so much on research nor data analysis.

The solutions are not extensible to the entire agricultural sector. There are sectors like the wine that may be more appropriate, especially in the case of producers who are looking for higher quality performance. In these subsectors it's more likely that investment in information technology would improve the wine production.

We want evaluate the use of wireless sensor networks technology for precision viticulture. First we intend to analyse the specific operational characteristics of a vineyard, section II. Second, we will propose the most appropriate architecture for a wireless sensor network in this environment, section III. The implementation of wireless sensor networks technology can take many forms depending on the environment, and objectives.

In addition, most of the proposed solutions have a high dependency between levels of the protocol stack, in order to optimize aspects such as energy consumption. The energy conservation is a main aspect for the viability of a wireless sensor network. That's why, we decide to realize an analysis per level; physical level, link, going through network level, and application level. The objective is to determine which solutions are the best for the operational environment under studio, which of them could be integrated optimally and, if necessary, propose improvements or new solutions.

Third, after analysing the operational environment and extrapolating different functional requirements, we choose ZigBee as radio technology. With ZigBee we have determined that the best option is to use a tree network topology and hierarchical routing. We have analysed the limitations of hierarchical routing and the assignation mechanism of identifications. After studying several alternatives, we have proposed our own solution to avoid the limitations of ZigBee hierarchical routing mode, section IV.

Fourth and finally we propose a deployment mechanism of the wireless sensor network based on pre-location of nodes and pre-assignation of addresses in order to get a optimal network structure, section V.

*Author to whom correspondence should be addressed; Universidad Carlos III de Madrid. Avda de la Universidad, 30 E-28911 Leganes (Madrid) Spain ; E-Mail gregorio.corral@uc3m.es; Tel.: +34-916-249-959; Fax: +34-916-248-749

2. Operational environment of a vineyard

The operational environment have a great incidence over requirements and design decisions. The vines position in the area under cultivation is fixed along the year, so our sensors will require no mobility. Initially we will not consider the mobility of other agents (operators, machinery, animals) inside the vineyard. We consider that the nodes position is known and will not change. The nodes position will be registered during the deployment phase. The vineyard extension can vary in the order of a few hectares to hundreds of hectares per parcel. Usually a winery has its vineyard spread out on different parcels. We will take 24.5 ha as reference size parcel. Also we assume a vertical trellis system instead of the more traditional goblet vine training system.The vertical trellis system is a structure formed by several rows of wire supported by poles. The vertical trellis allows a vertical driving of the vineyard vegetation. On the one hand, the vertical trellis system is able to maintain the plant and its fruit above ground, avoiding the humidity and in this way the promotion of fungi and several diseases on the vineyard. Moreover, the vertical trellis system facilitates the action of sun's rays on leaves and bunches. It is possible using the trellis system as support to deploy of a drip irrigation system controlled by sensors.

We considered several magnitudes to be measured, typically; temperature, humidity, solar radiation level, or ground ph. The fundamental magnitude is the humidity level of the vines throughout the year. The humidity is a factor that directly affects the production and quality of the crop. There are not strict real-time constraints regarding the collection of data, but we needed enough data to allow us to follow and explain the evolution of a phenomenon. We considered that the magnitudes measured have a linear behaviour, so that it is possible to extrapolate values from the collected data.

The requirements of the environmental sampling protocol (for the study and evolution of soil characteristics for example), determine the number of sensors available, location and density [2] [3]. To evaluate these factors we must take into account both the measurement surface and the communication coverage [4]. As for fault tolerance, the fall of nodes should not affect the completion of the tasks for which has been deployed the wireless sensor network [5]. In order to improve the network fault tolerance we can deploy redundant nodes and extend the life of these. The energy factor has a great impact on the service life of each sensor. We must keep a balance between reliability and cost to determine the number of nodes to be deployed. It is expected that technology becomes cheaper, but today sensors are expensive devices.

Energy is a limited and critical resource in the field of wireless sensor networks. There are two basic strategies to follow when you are constrained by a resource. On the one hand, you can find additional ways to power supply. In this case we can explore sources of energy from the environment, which has been called energy harvesting [6]. This possibility has been explored in the environment of viticulture so sensors could be powered by more than one type of energy source [7], for example; photon energy by solar panels, kinetic energy through movement of wind or water running along a pipe. On the other hand, efforts has been made in research, and design of energy efficient systems, proposing new mechanisms at link level and network protocols [8].

As we previously denote several routing protocols have been proposed [9] to improve the energy consumption. We deal with the issue and assume that if the solutions based on sensor networks are so specific of its scope, then we can go one step further and define a specific protocol for a specific solution.

We can use specific features of our operational environment to design the protocol. For example, we propose a routing algorithm based on the availability of energy, in the sense of other raised solutions [8], and also on its future availability. That is, predict the energy availability in a node based on its ability to recharge (for example via photon energy or kinetic). We can establish control systems that allow calibrating the energy consumption of the nodes of the network of sensors depending on the growing season (throughout the year), the climatic and environmental evolution (throughout the year, a month, a week or a day), or technical and functional needs. We can take advantage of the redundancy [9], both in terms of nodes and data. We can identify areas of adjacent nodes that provide redundant information. Having been identified these areas, we can use this redundancy to save energy by using techniques such as data aggregation [10]. These areas could be dynamically increase or decrease depending on the variation of the measured data. We could schedule idle periods for these nodes within each zone. The goal is that our sensor network could have a lifespan of at least one year or the corresponding period to a whole vineyard annual cycle [11]. This is an ambitious goal considering previous experiences [12].

One aspect to consider in our scenario is the validation or protection of the data itself. If some of the sensors forming the network is compromised in some way then the received data may not correspond with reality. When the communication between sensors or between the base station and sensors is compromised then we are in a cryptographic issue. But if the sensor itself is manipulated, for example to record a different temperature, this is solved by applying outlier detection techniques [13]. We could use paradigms like RANCAC (Random Sample Consensus) applied in other fields such as computer vision, in which it is assumed that a certain number of samples taken are committed [14].

2.1. Vineyard annual cycle

The lifetime of a vine can reach hundreds of years but the optimum productivity of each plant can be maintained for about fifty years. Each year the harvest marks the end of the growth cycle, and usually correspond with a year. Throughout each year the crop goes through different phases with particular characteristics. Each phase of growing can vary in time and months from one to another wine region of the world, but the phase sequence are the same. In this case, our geographical reference is the north central Spain (Rivera del Duero, Toro or Rueda wine regions).

During the first growth phase the vine remains dormant. The shoots are hardened, without leaves, and sap accumulates in the stem. This dormancy phase lasts from November to about March following year, coinciding with rising temperatures. In this phase the water needs are limited. The work in field include pruning the branches and the adequacy of the vineyard for the next season. This would be the best time for a initial deployment of the network of sensors or to check the installation, in the case of already being deployed.

In March the vine awakens from its dormancy. The first sign that the vine is awaken is when through cuts, produced during pruning, begins to leak a colourless liquid, sometimes reddish, the so called *weeping of the vine*. This phase usually lasts about three weeks.

In April, the *sprouting* begins when temperatures reach ten Celsius degrees. Buds swell, the scales that protect the buds open it and a small initial leaves appear. During this time and during subsequent flowering phase the vine is vulnerable to frost damage. In this case, detect abrupt fall in temperatures could justify an alarm system supported by a wireless sensor network.

In late May *flowering* occurs. The *flowering* consists in the flower opening, freeing pollen that falling on the stigmas and allows fertilization. Sometimes due to lack of heat, excessive moisture or lack of vigour of the plant, the flower is not correctly fecundated. The result is that bunches will have few berries. Then it is said that there have been a *flowering shift*. During sprouting and flowering rarely there are water shortages because water demand is low. Water deficits during the flowering stage reduce the potential harvest by reducing the number of bunches.

Arriving July, the vine reaches its physiological maturity. The fruits still are far from be mature grapes, but has already started the way to the harvest. In July, begins a phase called *veraison*. In veraison, the grapes stop being greens to become yellowish for white grapes and pink for red grapes varieties. In addition, the grape begins to lose acidity and accumulate sugar.

In September, the berries are very sour and become mainly in sugary, the grape skin gradually softens and the red grape takes an intense colour. This is the phase of maturation whose ending is difficult to define. The phase of maturation ends with the harvest.

October usually is the month to harvest. The winemaker decides when to make the grape harvest, depending on the wanted type of wine. With early harvesting, the wines are fresh and green. With delayed harvesting, the wines use to be high in alcohol and colour.

The most critical period from hydric point of view starts from flowering, late May, to veraison phase and just before grape harvest. At first, the needs of water of the vine get bigger to produce fruits. In the maturation phase the amount of water has to be enough, but not too much, to achieve balance between quality and productivity. These phases correspond with the periods of major water shortages and increased variability of humidity levels within a crop [15].

On November the cycle is closed, even before grape harvest the vine walks to his exhaustion. Begins the path of the vine to its hibernation.

In general we can conclude that during the months of November, December, January and February our network will have a reduced activity and we will not need gather a exhaustive data collection. The activity should gradually increase over March and reach its peak during the months of increasing water shortages.

2.2. Spatial variability of data

It is well known by vintners that there is a variability of environmental characteristics (temperature, humidity, solar radiation level, or ground ph) into a crop, but single treatment for all is applied. This variability has already been observed through experiences based in data collection with sensor networks [4]. As for weight yield of grapes, for example, can be the case differences of up to 10 times higher productivity from one area to another in the same plot [16]. More important is the variability of grapes maturation parameters.

The winemakers want to get grapes with uniform maturity because this is a factor that directly affects the quality of the wine. The evolution of the humidity of the plant throughout a whole growing season, has a direct impact on the yield and quality of grapes. Spatial variability of moisture levels recorded throughout a year on a crop has been empirically observed. Studies on this matter have found that in periods of year where water restrictions are greater the spatial variability is greater, there is a greater diversity of values and granularity of them along the vineyard [15]. This means that with a small water restriction less samples are necessary to follow the evolution of vineyards. On the contrary, with greater needs of water, the density of samples to take must be greater. From these studies it also follows that, in case of water shortages, the next factor that has impact on the moisture of the vine is the type of terrain, followed by the grape variety. The wireless sensor network and precision viticulture technologies, enable following the vineyard evolution. We can compose a knowledge database and use this data to support specific decisions and actions.

3. Wireless sensor network topology

The wireless sensor networks usually are composed of at least two node types; sensors for data collection and one or more sink nodes for receiving all the information collected by each of the sensors in the network. The nodes can form a homogeneous network, all capable for data collection and its forwarding, or we could have specialized nodes, either for collecting certain data or forwarding information.

3.1. Environmental restrictions

The objective is to collect data to assess the variable characteristics of the cultivated soil and vine itself. It could be thought that it is a good idea deploy a sensor per vine. Modern vineyard parcelling depends on soil quality. If the soil is very fertile, then more vines can be planted per hectare while in poor soils the vines density will be lower. We assume a vineyard parcelling with 2.5 m to 3 m of distances between rows and 1.5 m to 2.5 m of distance between vines in each row. This vineyard parcelling involve a deployment of sensors in a mesh structure of about 3 m distance between nodes. This assumption may not be feasible in terms of cost. It is not necessary such a density of sensors because the values taken at different sampling points could have a strong spatial correlation [17], so that you can apply statistical methods of multiple regression analysis to infer intermediate values [18], although the closer are sampling points each other, the more accurate are the inferred data. In any case, it is more efficient identify growing areas with similar characteristics, and apply the same treatment on them [16].

Several studies about mesh sampling techniques have established that the distance between sampling point must not be more than 100 m [17]. The distance between sampling points depends of the magnitudes to be analysed. For example, for the study soil fertility the following parameters are usually measured; pH level, phosphorus (P), potassium (K), calcium (C), magnesium (Mg), carbon (C) and nitrogen (N). In the case of a crop treated with inorganic fertilizer the restrictions are not very high, but the study determines that for soils treated with organic fertilizer is necessary at least 15m of separation between nodes [17].

In a experience to control the levels of evaporation and transpiration of the vineyard [12], the collected measures were temperature, humidity, and radiation level. For this scenario 22 sampling points were used on a surface of 10.50 ha. The distance between sampling points was approximately 24 m, although the measures were taken at different heights (10 cm above the ground, at 1 m and 1.60 m), so 66 sensors were used.

In addition, sensors have to survive long enough and, during this time, the network must maintain a minimum level of integrity that allows data to reach their destination. Again, these parameters depend of how efficient is the energy consumption of our network. The topology, or location, of nodes is a decisive factor. Moreover, the position of our sink node directly affects consumption of the network. Analysing different topologies has been determined that the most efficient solution, from the energy consumption point of view, is when the sink node is at the centre of the mesh [12]. This is not always possible, from a functional point of view. The most common location of the sink node is in one of the network edges.

In a homogeneous network all nodes have the same capabilities and functions. The nodes close to the sinks have higher overall energy consumption, that's because these nodes concentrate much of the traffic towards the sinks. On one hand, there is a risk that the lifetime of those nearest nodes to the base stations will be drastically reduced. On the other hand, these nodes could stay disabled, leaving some other network nodes without connectivity. You can design the network protocol so that the forwarding decisions would take into account not only criteria of proximity but power capacity too. From a point of view of network topology, you can choose to deploy specialized nodes into data relay [19]. These extra nodes will not be distributed evenly, but they will be located to reinforce network areas with a highest concentration of traffic. Also we can use more than one sink node. The existence of more than one base station at different locations at the crop edges would give us flexibility to redirect traffic to one or another, depending on the availability of energy.

3.2. Topological features

The tendency is to choose between three radio technologies; IEEE 802.15.4/ZigBee, 802.11 (Wi-Fi) or 802.15.1 (Bluetooth). The 802.11 option enables very high rate data transfers, but involves high energy consumption, at least with regard to wireless sensor networks. The Bluetooth (802.15.1) option have a limitation in number of nodes that can form our network. 802.15.4/ZigBee has been specifically developed to meet the needs of wireless sensor networks. ZigBee is a standard bidirectional low-power wireless communications, developed by ZigBee Alliance to be integrated into all types of electronic devices in various fields.

There are different versions and extensions of ZigBee. The first specification was ZigBee 2004 which was replaced by ZigBee 2006. At present the specification adopted by the market is ZigBee PRO (or ZigBee 2012). The Green Power extension takes into account the presence of self-powered devices, or devices that enable energy harvesting. The integration of these devices as part of a ZigBee network is oriented to optimize power consumption of the whole network. Other standards proposed by ZigBee Alliance are ZigBee RF4CE and ZigBee IP. ZigBee IP is a IPv6 based wireless mesh network proposal in which the devices have direct Internet access. On other hand, ZigBee RF4CE standard is designed to compete directly with Bluetooth. ZigBee 3.0 standard in currently under development in order to unify the different ZigBee standards and its extensions into a single standard. In this document we assume the ZigBee PRO standard as reference.

The new ZigBee modules operate in the 2.4 GHz band, following the 802.15.4 standard. The use of 2.4 Ghz band allows assembling internal antennas and, in this way, the nodes are more easy to handle. We obtain rates of 250 kbps and coverage of at least 125 m. Actually, the coverage is less because the foliar mass of vines has the ability to absorb microwaves. Also the signal is reduced due to plastic housings and some vine training systems. Those training systems use metal structures to hold plants, this structures induce many signal reflections. Under these conditions the transmission range is reduced to 20 or 25 meters [20].

The 802.15.4 standard defines two types of nodes: the so called reduced function devices (RFDs) and the full function devices (FFDs). The RFDs act only as end nodes in the network and are equipped with transducers. The FFDs perform functions of coordination of network nodes but can also be end nodes with associated sensors. FFDs generate synchronization signals, provide communication service and subscription service . The ZigBee specifications consider three possible network topologies; Star, tree or mesh. With a star topology one FFD node assumes the roll of PAN (Personal Area Network) coordinator. The RFDs and other FFDs can only connect through the PAN coordinator. The star topology is an option when minimizing data transmission delay is critical for the application-level, although the coverage area of the network is limited to the transmission coverage area of the PAN coordinator.

The point-to-point topology allows any FFD coordinator node to communicate with any other coordinator who is in its coverage area. A special case of point-to-point is the tree topology. The tree topology proposed by the 802.15.4 standard is a tree rooted at the PAN coordinator. The tree structure is established by parent-child relationships at MAC level. Once generated, the tree structure is fairly static, and routing decisions are based on parent-child relationships established at MAC level. In tree topology, once formed the network, the routing is quickly established and create routing tables is not required [21]. The tree topology is more efficient from the point of view of energy consumption because of working in beacon mode, as we will discuss below [22]. The tree topology disadvantage is its lack of flexibility, it is very sensitive to possible falls of coordinator nodes. A second variant of point-to-point topology is a mesh structure. The mesh topology allows communication between any FFD, without the need to having a parent-child relationship. The mesh networks are much more flexible in case of node falls but they require the establishment of routing tables and are less efficient from the point of view of energy consumption [22].

A FFD must be chosen as PAN coordinator. This coordinator node can work on beacon mode and use a superframe to support a slotted CDMA-CA protocol. In beacon mode, an inactive interval is defined within the superframe, and during this interval the coordinator PAN can stay asleep. An active period and another inactive will be assigned to each router within each superframe. The end nodes will be on sleep mode most of the time and are activated periodically. The tree topology, defined on the ZigBee specification, allows working in beacon mode. The instants at which the nodes awaken or entering on sleep mode must be fixed so that each router has to wake twice per cycle. First time to receive packages, from children nodes, and a second time to forward packets. There are authors who proposed algorithms to schedule active periods within each superframe for each router in coordination with their children. This is in order to improve the data forwarding along the hierarchical tree [23]. The second mode on which a coordinator PAN can work is without superframe. On this mode it uses an not slotted CDMA-CA mechanism and the coordinator is always active.

It has been studied the effectiveness of using or not using of superframe. Notice that in 802.15.4 the exchange of RTS/CTS packets are not contemplated in the same manner that in 802.11. The exchange of RTS/CTS packets is used to avoid the hidden terminal problem when working with multi-hop networks. This is the case of three nodes placed so that two of them are not directly reachable each other. It may happen that these two nodes try to send information simultaneously to the third node, then a collision occurs. It has been determined that the not slotted mode makes a better use of the channel, but do not have mechanisms for energy savings and neither ensures data delivery in a specific period of time.

A compromise solution must be adopted between improving rate of data transfer and maximizing the lifetime of each sensor. Our operational environment has no real-time constraints, and it does not need to use intensively the channel. We have chosen a solution based in improving the energy consumption for extending the life time of the network. That is why the most optimal choice for our application is to use the slotted mode, although, working in slotted mode the idle and working periods must be properly calibrated.

The slotted CSMA-CA mode allows transmission of beacon signals from the FFDs of the network, in order to synchronize communications. These synchronization periods imply contention periods and energy overload due to collisions. With high data transfer rates and dense networks, the probability of collision is higher and increase the power consumption. In our environment we estimate that the data transfer rate is not high. The network should not be very populated, both from point of view of economics cost as from a functional point of view.

3.3. Network Addressing

The functionalities that include the specifications of ZigBee network level are; multi-hop routing, route discovery and maintenance, security, join and leave from network, in addition to the allocation and management of 16-bit addresses [24].

At the beginning of a joining process each node makes use of link layer services to discover the routers neighbouring nodes that are advertised. Each node maintains a table with their neighbours, nodes within its transmission range. Each node chooses a network (multiple networks can overlap using different channels), selects a parent node and joins to it. This parent node is selected among neighbouring nodes registered on its neighbour table. The neighbour table also holds information on the status and quality of links. When the parent node receives the association request, through its link level, it assigns an address to his new child node and completes the association. This manner a tree structure is generated based on parent-child associations, where the root will be the network coordinator node.

The coordinator node sets three parameters; maximum number of children of type router that can have each node (R_m), maximum number of children that can have each node (C_m) and the maximum depth of the tree (L_m). Each parent calculates a consecutive number for each of its new children. To calculate this number the parent uses R_m , C_m, L_m, and the depth of the new node added to the network (d). This number is assigned to the new node as network address ($C_{skip}(d)$). If the new node is not a router, the parent uses the following equation to calculate the address:

$$C_{skip}(d) = \begin{cases} 1 + C_m(L_m - d - 1), & \text{If } R_m = 1 \\ \frac{1+C_m-R_m-C_m*R_m^(L_m-d-1)}{1-R_m}, & \text{otherwise.} \end{cases}$$

If the new node is a router, then the parent use the following equation to calculate the address:

$$A_n = A_{parent} + C_{skip}(d) * (n - 1) + 1$$

A_n is the direction of the n-th new router node of node parent A_{parent} with depth d. On the other hand, If the new children node is an end node, then the address will be assigned using the following equation:

$$A_n = A_{parent} + C_{skip}(d) * R_m + n$$

Where A_n is the address to be assigned to the n-th children node with depth d. If it uses this model only a coordinator or

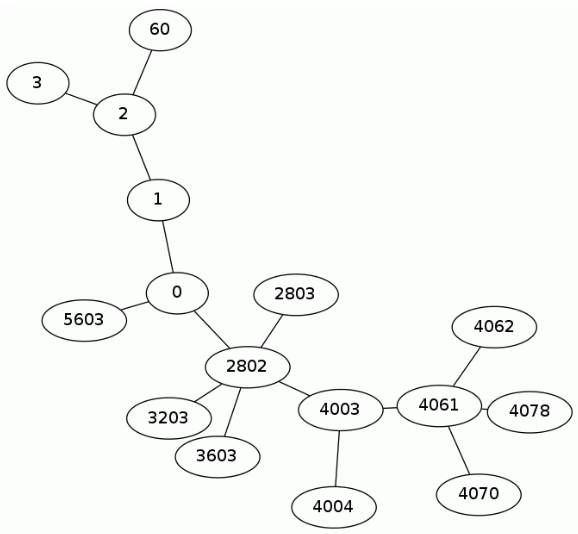

Figure 1: Example of addressing for $R_m = 7$, $C_m = 7$, y $L_m = 5$

router node can assign addresses. It may happen that a router node run out addresses, in this case, the child node has to find another parent among its neighbours. If the new node does not find another parent then it can not join to the network.

For example, for $R_m = 7$, $C_m = 7$, y $L_m = 5$ the values for $C_{skip}(d)$ will be:

d	C_skip
0	2801
1	400
2	57
3	8
4	1
5	0

For these values and a root node with $Addr = 0$ the assigned addresses are shown in Figure 1.

The distributed address assignment mechanism described in ZigBee specification does not guarantee the build of a balanced tree. In Addition, it may happen that a router reaches the limit of childrens while another may have not. With the distributed address assignment mechanism the best case is when the coordinator node is located in the middle of the network. In this case the diameter of a network is determined by L_m and will be $2 * L_m$, though the value for the diameter may be much lower. The ZigBee specification also allows a stochastic method to assigning addresses but it could happen that the same address will be assigned to two different nodes.

To solve the problem that a parent remains without addresses, while another node may have free addresses to assign, some authors have suggested additional mechanisms of address assigning and routing [25]. For example, a mechanism to allocate unused addresses, and previously pre-assigned to others parent nodes. Also, defining tables for each node where to store information about address reallocation [26]. The information stored in these tables is useful for routing because it allows locating nodes with assigned addresses. Applying these mechanisms can improve the address space use, but without considering the problem of generating unbalanced tree structures. The efficiency of dynamic routing procedure to use the address space depends partly on the geographical area where the network is deployed. The optimal case is a square area where we can deploy a network with regular topology. The network topology is determined by the values of R_M, C_m, L_m. The topology is regular when the diameter of the tree is regular with respect to its depth. While the worst case occurs with an elongate rectangular area. In most cases the deployment area geometry will not be the most optimal. An interesting proposal is subdividing the area of deployment into areas of optimal proportions [27]. The areas and the topologies applicable in each area would be defined before the deployment of the network and special nodes would be designated to work as bridges between different subareas.

The sharing out of the address space available has an additional constraint. The restriction we found is that the ZigBee specification defines a network address of 16 bits, so our address space is $[0, 2^{16} - 1]$. Initially 2^{16} nodes are enough for almost any sensor network. The problem is that distribution of address space, according to the ZigBee specification, is very inefficient. For example, considering a scenario with $R_m = C_m$, the address space N to distribute from root node (with depth $d = 0$) among its R_m will be:

$$R_m * \frac{1 + C_m - R_m - C_m * R_m^{L_m - d - 1}}{1 - R_m} = N;$$

$$R_m * \frac{1 + R_m - R_m - R_m * R_m^{L_m - 0 - 1}}{1 - R_m} = N;$$

$$R_m * \frac{1 - R_m^{L_m - 1 + 1}}{1 - R_m} = N;$$

$$\frac{R_m - R_m^{L_m - 1 + 1 + 1}}{1 - R_m} = N;$$

$$\frac{R_m - R_m^{L_m + 1}}{1 - R_m} = N;$$

$$R_m - R_m^{L_m + 1} = N * (1 - R_m);$$

$$-R_m^{L_m + 1} = N * (1 - R_m) - R_m;$$

$$R_m^{L_m + 1} = R_m - N * (1 - R_m);$$

$$R_m^{L_m + 1} = R_m - N + N * R_m;$$

$$\ln(R_m^{L_m + 1}) = \ln(R_m - N + N * R_m);$$

$$(L_m + 1) * \ln(R_m) = \ln(R_m - N + N * R_m);$$

$$L_m \ln(R_m) + \ln(R_m) = \ln(R - m - N + N * R - m);$$

$$L_m \ln(R_m) = \ln(R_m - N + N * R_m) - \ln(R_m);$$

$$L_m = \frac{\ln(R - m - N + N * R_m) - \ln(R_m)}{\ln(R_m)};$$

$$L_m = \frac{\ln((1+N)*R_m - N)}{\ln(R_m)} - 1;$$

If we said that the maximum address space we use is $N = 2^{16}$ and for instance, we set $R_m = 7$.

$$\frac{7 - 7^{L_m+1}}{1-7} = 2^{16};$$

R_m	L_m
2	15.00
3	9.72
4	7.79
5	6.75
6	6.08
7	5.62
8	5.26
9	4.99
10	4.77

The highest depth of our tree will be 5, which is insufficient for a network deployed in a 10x10 mesh structure. We will never get a graph with 100 network nodes connected. Even doubling the size of our address space, the highest depth will be 11. In fact, the highest depth of a network with a tree structure, generated according to the ZigBee specification, is 16 [28].

An alternative to the distributed address assignment model is to apply a static scheme of addresses assignment. Some suggest a static assignment model for better use of address space [29].

3.4. Routing

ZigBee includes two routing protocols that can be applied in combination. ZigBee uses the tree structure generated from the process of dynamic address assignment as support for hierarchical routing. The routing is performed through parent-child link structures established during aggregation process of nodes. A router can easily determine if a message is addressed to one of its descendants, otherwise the message is forwarded to its parent node. If the router address is node A, d is the depth of the node in the hierarchical tree, and the destination address D satisfies the following condition:

$$A < D < A + C_{skip}(d-1)$$

Then the destination node is a descendant of the node with address A. If D is the address of an end node and meets the following condition:

$$D > A + R_m * C_{skip}(d)$$

Then the next hop address N is equal to D, $(N = D)$. If the condition is not met, then N is calculated as follows:

$$N = A + 1 + \left\lfloor \frac{(D - (A+1))}{C_{skip}(d)} \right\rfloor * C_{skip}(d)$$

Based on the hierarchical routing mode, defined in the ZigBee specification, several improvements have been proposed. These proposals try to explore the possible shortcuts between nodes which form the tree structure. One of these proposals is the ETREA protocol [30] for discovery of optimal alternative paths. ETREA takes into account the number of hops between source and destination as well as the residual energy of the nodes that form part of the route.

The alternative is to use ancestors in order to release load from parents. The idea is to combine the hierarchical structure and the neighbour table information to choose a node and jumping directly to the grandfather or to explore different branches.

The second routing mode enabled by ZigBee is a point-to-point routing mechanism over mesh topology. This second routing mode is more complicated and usually does not operate in beacon mode. ZigBee allows the application of a simplified version of the AODV protocol to implement this point-to-point routing mechanism [21].

The hierarchical routing is more efficient in the case of a bursty data transmission model. AODV is more efficient when the data is generated continuously. The ZigBee specification for the network layer of each node, called NWK, allows a node to apply the hierarchical routing based on the parent-child relationships or use a route discovery algorithm based on AODV.

4. Gateway mode to extend network size

Our proposal to avoid the limitations of hierarchical addressing of ZigBee is subdivided the tree into subtrees [31]. We have implemented a mechanism that allows nodes to make functions of gateway between different networks. When a router node joins the network with the maximum depth and the router is enabled to work in gateway mode, it generates a new network ID and announces its presence as the root node of a new tree. The gateway will forward the packages received from their childrens towards its own root node. This model avoids depth limits but complicates the routing from root node to others nodes that are below gateway. We consider that in our operating environment most of the traffic will go towards sink node, which typically coincide with the tree root node.

The figure 2 shows an example of association of nodes using the gateway mode. In this example the maximum depth is set to 3 ($L_m = 3$) the initial PANid is 9. Nodes 125 and 126 of this tree make a bridge between the subtrees with PANid 1 and 6 respectively.

It is also possible to deploy a solution based on Content Centric Network (CCN) [32] to expand the address space and route packets from root node to children through one or more gateways.

Even if we use the model of gateways between subtrees, it is better to use the address space as much as possible before generating a new subtree. This is why it is better to choose the lower deep node of the tree as parent. We will encode the node depth using the four most significant bits of PANid. Each node will use this information to choose his parent.

The figure 3 shows an example of association using the gateway mode and giving priority to nodes located at lowest depth. The maximum depth is set to ($L_m = 3$), the initial PANid tree 9, nodes 125 and 126 go to make a bridge between the subtrees

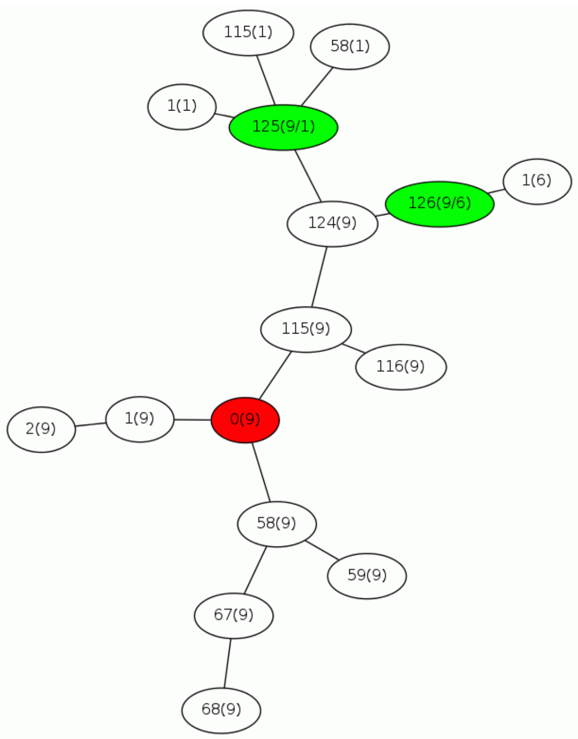

Figure 2: Gateway example with $R_m = 7$, $C_m = 7$, and $L_m = 3$

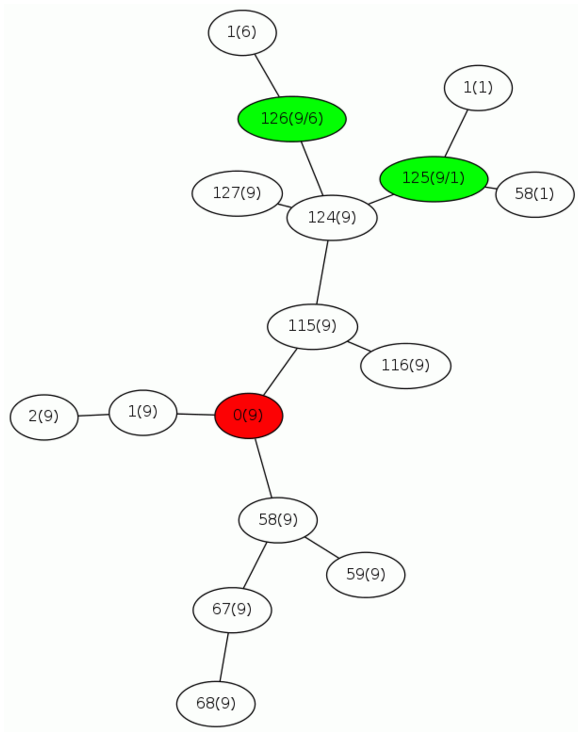

Figure 3: Example of gateway mode with improving of address space use for $R_m = 7$, $C_m = 7$, y $L_m = 3$

with PANid 1 and 6 respectively. In the previous case a node was associated with address 115 to the PANid 1, in this case the node gets the address 127 and is associated to the top level tree with PANid equal to 9.

5. Wireless sensor network deployment

We propose a static addressing mechanism. This addressing mechanism must supports hierarchical routing over a balanced tree structure.

A second proposal is to make an analysis before deploying the wireless sensor network. We will use the result of this analysis to calculate an optimal tree and assign addresses. The resulting tree should be balanced, with a minimum depth and cover all crop area with sensors.

Given the topological characteristics and the application environment we model the crop area as a plane covered with a mesh structure of sensors. This mesh will match the geometry of the vineyard parcelling. Given the fixed location of nodes, a pre-deployment operation will be performed. We look for the highest probability of connectivity between nodes of the network, and assuming a maximum coverage of 20 meters, we will set a maximum horizontal and vertical spacing of 12 m between nodes. This guarantees a minimum number of neighbours, between 3 (nodes located on mesh corners) and 8 nodes (nodes that are at the centre of the mesh). We used OMNeT [33] for implementing and simulating the association process of nodes. We have got samples of the potential tree-like structures that can be generated, as well as their characteristics. To characterize these structures we generated a hundred of samples per scenario, a hundred of different trees in a 17x17 grid structure (289 nodes and a sink). From these samples, and for each of the scenarios, we studied the distribution of children per node, the depth distribution of nodes within the structure, as well as the depth of leaf nodes in the tree.

5.1. ZigBee association without activation delay

The network nodes were associated to form a tree, as we have discussed previously. The tree topology is the most efficient organization from an energy point of view. According to the association mechanisms that follows the ZigBee specification, each node chooses a parent among its neighbours for joining it to the tree. In an ideal first scenario, figures 4 and 5, all nodes are activated simultaneously. In this first scenario, the choice between neighbours to become parent is random.

5.2. ZigBee association with activation delay

In a second scenario, figures 6 and 7, we introduced a node activation with random delay (between 0-2 minutes). Even though the results are similar to the simulations without delay, this is a more realistic scenario. We have tested the behaviour of the ZigBee association mechanism on such environment.

5.3. ZigBee Association with delayed activation and prioritizing by depth

In a third approach, Figures 8 and 9, we have included a variation in the node aggregation mechanism. In this third test, the nodes with lowest depth, closer to the sink, are designated as best candidates as parents.With this modification we expect

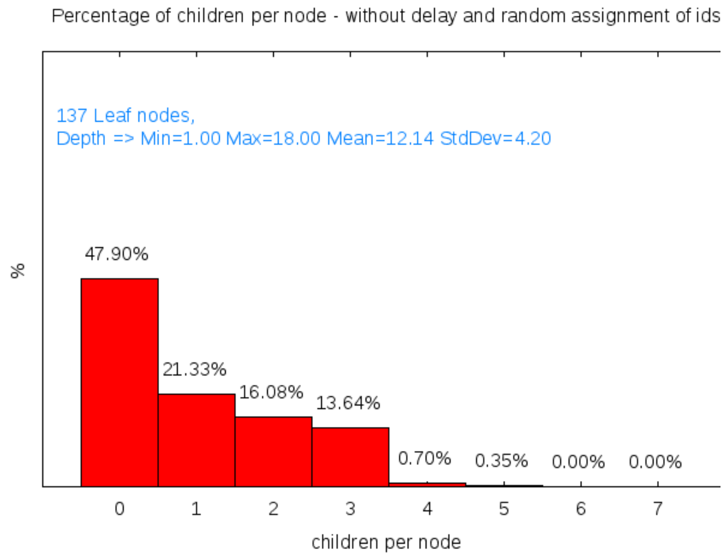

Figure 4: Distribution of children per node without delay and randomization

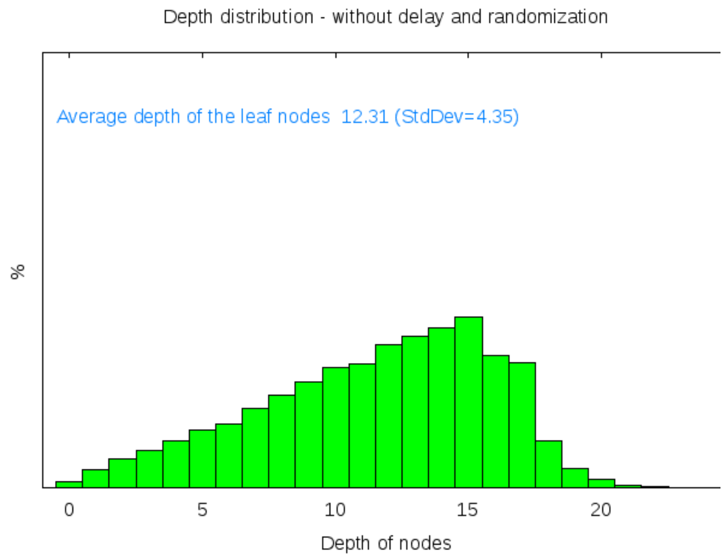

Figure 5: Node depth distribution without delay and randomization

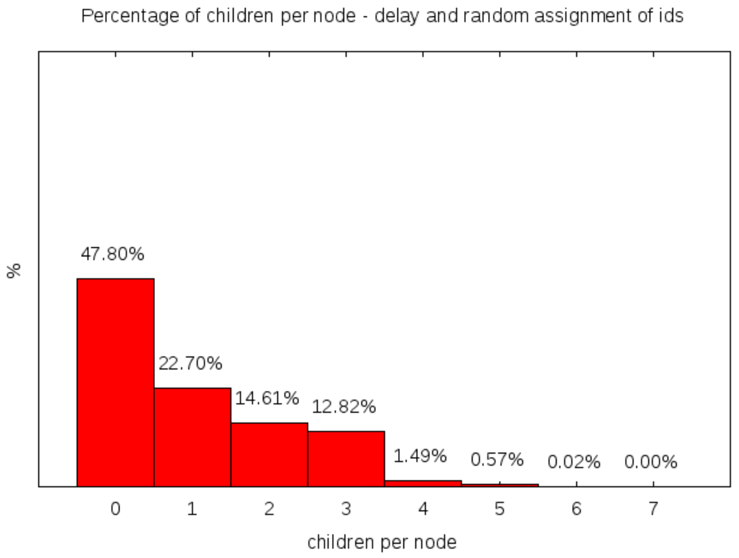

Figure 6: Distribution of children per nodewith delay and randomization

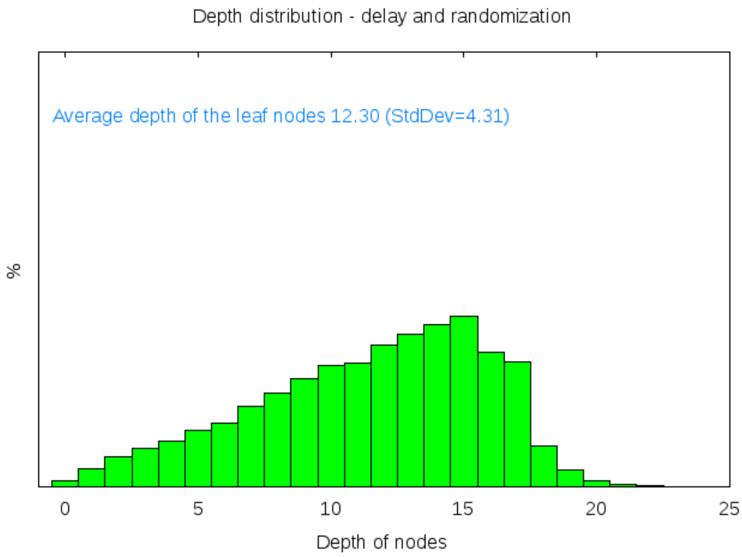

Figure 7: Node depth distribution with delay and randomization

a structure more horizontal, although there are no significant variations between the graphs, we have tested a variation over the ZigBee association mechanism.

5.4. Creating tree of associated nodes based on Dijkstra

The three first scenarios are simulations of ZigBee aggregation process. Now we test building a tree over a grid structure. To build this tree we designate a root node and we apply different methods for generating trees. First we apply Dijkstra's algorithm (see figure 12) to generate the minimum spanning tree, figures 10 and 11.

We apply the basic algorithm on two variants. The first variant (see figure 13), figures 15 and 16, is to make a random choice between two paths with the same cost. This random choice is made, while tree is being build, in each iteration of the Dijkstra algorithm. The objective is to generate a poll of minimum spanning trees that could be applied.

The second variant (see figure 14), figures 17 and 18, intend to balance the load among nodes, so that, between two paths with equal length, the best candidate is the path that begins with the node with fewer children.

With Dijkstra significantly increases the number of nodes with one child, on the other hand the two Dijkstra modifications improve the generation of highly branched trees. With a modified Dijkstra the tree generated is flattened. The average height of end nodes is 12.

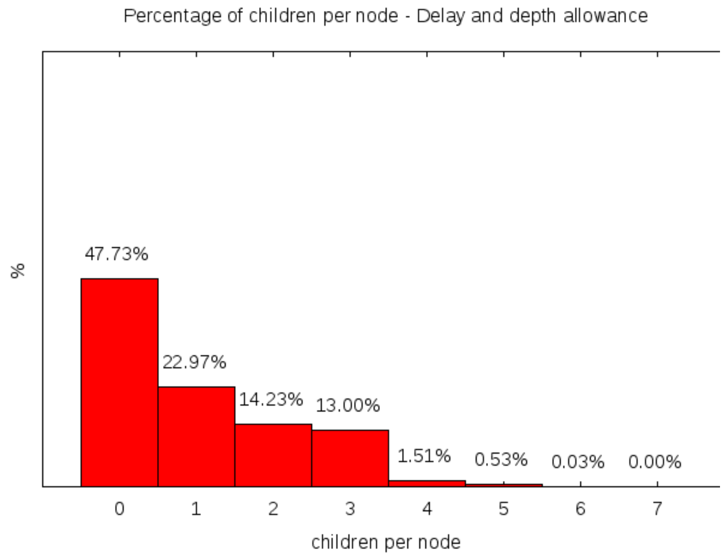

Figure 8: Distribution of children per node with delay and assignment by depth

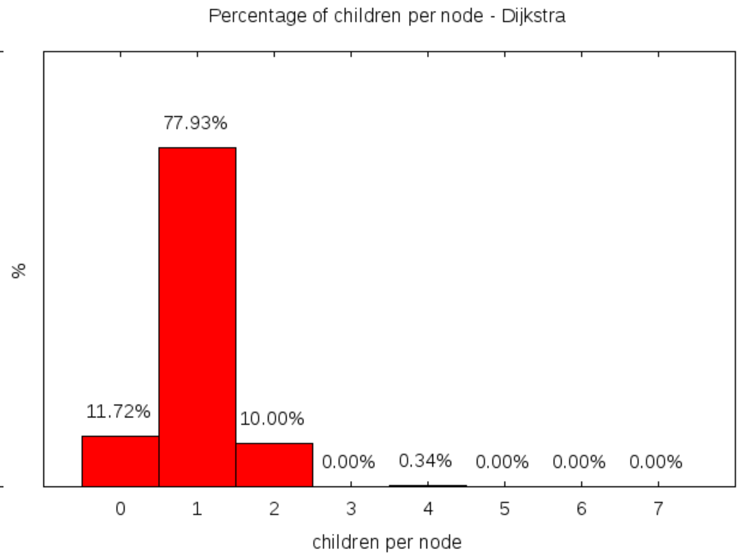

Figure 10: Distribution of children per node with Dijkstra

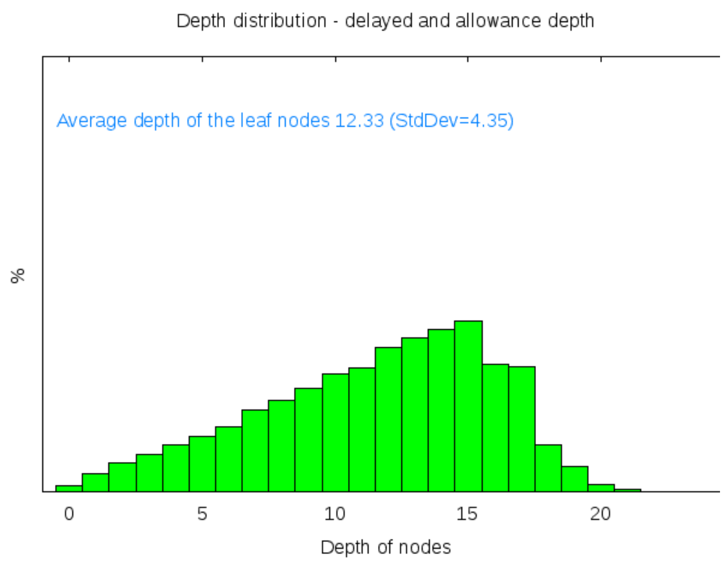

Figure 9: Node depth distribution with delay and assignment by depth

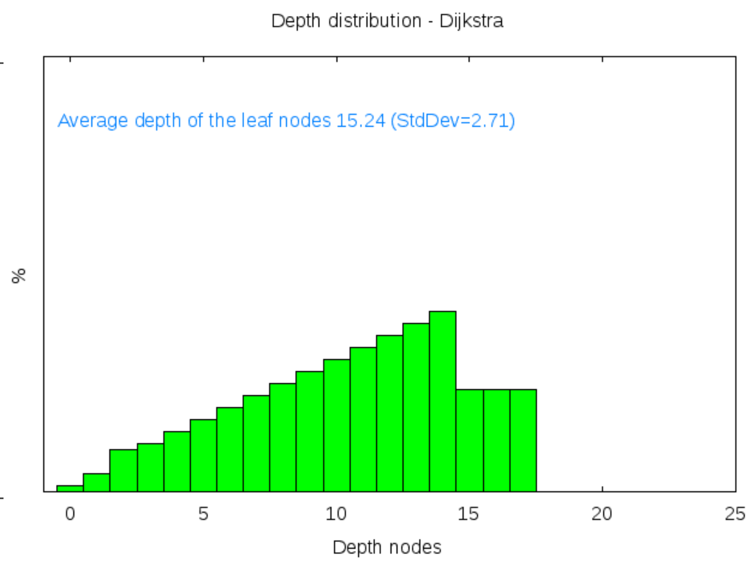

Figure 11: Node depth distribution with Dijkstra

5.5. Creating a tree of associated nodes with a cascade propagation model

The next option is to create the tree by applying a cascade propagation model, figures 19 and 20. This model follows the schema of propagation of information in a social network or transmission of an infection in a population [34]. To apply this model we have used the libraries of SNAP platform [35]. SNAP is a tool for defining and manipulation of graphs for network modeling. In the case of the cascading propagation model we applied a probability of 50% of spreading (or infection). This model reproduces a structure similar to the generated by ZigBee aggregation process. With this model we get a large number of end nodes.

5.6. Creating a tree of associated nodes based on the Kronecker product

Finally, we generate graphs using successive iterations of the Kronecker product [36]. We define an adjacency matrix to setting the probability of establishing a link between each pair of network nodes. In a first case, figures 21 and 22, using matrix 1, we have a similar model to the gotten by simulation of ZigBee aggregation process. In a second example, figures 23 and 24, using matrix 2, the results are similar. In both cases the depth of the nodes significantly increases.

$$\begin{bmatrix} 0.9 & 0.5 \\ 0.5 & 0.1 \end{bmatrix} \tag{1}$$

```
function Dijkstra(Graph, source):
  dist[source] <-- 0
  prev[source] <-- undefined
  for each vertex v in Graph:
     if v != source
        dist[v] <--  infinity
        prev[v] <-- undefined
     end if
     add v to Q
  end for
  while Q is not empty:
     u <-- vertex in Q with min dist[u]
     remove u from Q
     for each neighbor v of u:
        alt <-- dist[u] + length(u, v)
        if alt < dist[v]:
           dist[v] <-- alt
           prev[v] <-- u
        end if
     end for
  end while
  return dist[], prev[]
end function
```

Figure 12: Dijkstra Algorithm

$$\begin{bmatrix} 0.4 & 0.3 \\ 0.4 & 0.2 \end{bmatrix} \tag{2}$$

6. Conclusions

The design strategy of our wireless sensors network is to adapt the design to the specific needs of the application environment. Once the particular characteristics of the environment have been analysed, we have inferred some requirements and characteristics. We did not consider requirements of mobility, or real-time services. We needed a periodicity in the data collection that allows us to follow the evolution of certain values. We have determined that the critical parameter to be measured is the moisture level. We assume that the moisture level, and other parameters of the vineyard, are linear. So we can extrapolate values from the data collected.

A primary requirement is extended the activity and network lifetime to a full growth cycle of the vineyard, about a year. To extend the lifetime of the wireless sensor network is directly related with the optimization of energy consumption. In this sense, we propose adapting the network activity, and therefore energy consumption, to each stage of the growth cycle of the vineyard. For example, when the vineyard is in dormancy phase, the wireless sensor network also works at a low level of activity. Moreover, the activity of the network will be more intensive for critical stages of the growth cycle. The network activity is also adaptable spatially. From the point of view of the

```
function DijkstraRamdon(Graph, source):
  dist[source] <-- 0
  prev[source] <-- undefined
  for each vertex v in Graph:
     if v != source
        dist[v] <--  infinity
        prev[v] <-- undefined
     end if
     add v to Q
  end for
  while Q is not empty:
     u <-- vertex in Q with min dist[u]
     remove u from Q
     for each neighbor v of u:
        alt <-- dist[u] + length(u, v)
        if alt < dist[v]:
           dist[v] <-- alt
           prev[v] <-- u
        else if alt = dist[v]
          // At  equal distance
          // !! random choose
          change <-- ramdom[true ,false]
         if change:
            dist[v] <-- alt
            prev[v] <-- u
         end if
        end if
     end for
  end while
  return dist[], prev[]
end function
```

Figure 13: Dijkstra's algorithm with random selection of path

```
function DijkstraBalanced(Graph, source):
  dist[source] <-- 0
  prev[source] <-- undefined
  for each vertex v in Graph:
     if v != source
        dist[v] <--  infinity
        prev[v] <-- undefined
     end if
     add v to Q
  end for
  while Q is not empty:
     u <-- vertex in Q with min dist[u]
     remove u from Q
     for each neighbor v of u:
        alt <-- dist[u] + length(u, v)
        if alt < dist[v]:
           dist[v] <-- alt
           prev[v] <-- u
        else if alt = dist[v]
          // At equal distance
          // !! choose by
          // number of childs
        if childs(u) < childs( prev[v]) :
             dist[v] <-- alt
             prev[v] <-- u
         end if
        end if
     end for
  end while
  return dist[], prev[]
end function
```

Figure 14: Dijkstra's algorithm with load balancing

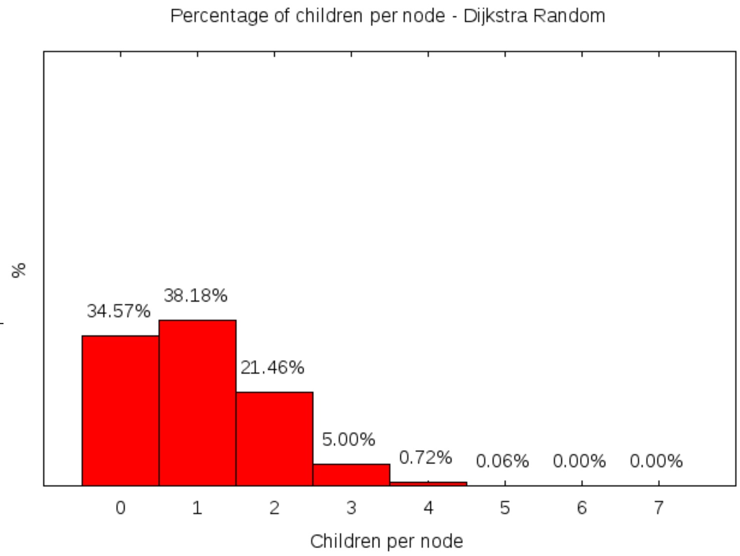

Figure 15: Distribution of children per node with Dijkstra and random designation

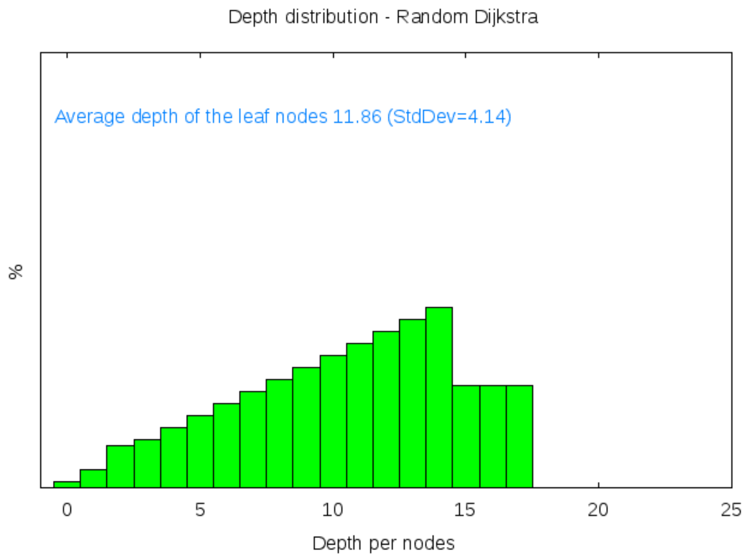

Figure 16: Node depth distribution with Dijkstra and random designation

farmer, it is more useful to identify growing areas, with similar characteristics, that specific points. Once identified these zones, it is only necessary to obtain a few representative samples, so the consumption is also optimized.

We have extrapolated a number of functional requirements from different studies and authors. In the worst case, the maximum transmission range, of a ZigBee node within a vineyard, will be about 20 meters [20]. We considered the structure of a classic vineyard parcelling for determinate the location of nodes. The nodes of the sensor network will be disposed in a mesh structure about 12 meters distance each other (considering the worst case scenario in terms of transmission range). We have determined, from some studies, that the tree topology is the most optimal from the point o view of energy consumption

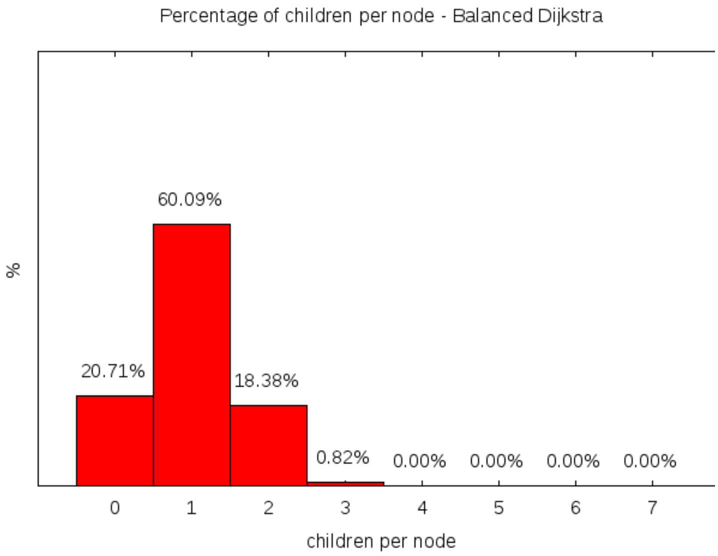

Figure 17: Distribution of children per node with Dijkstra and load balancing

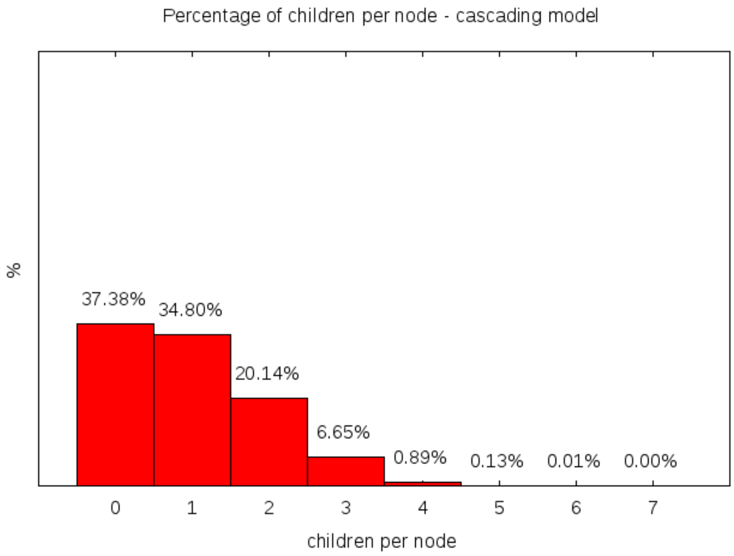

Figure 19: Distribution of children per node with a cascade propagation model

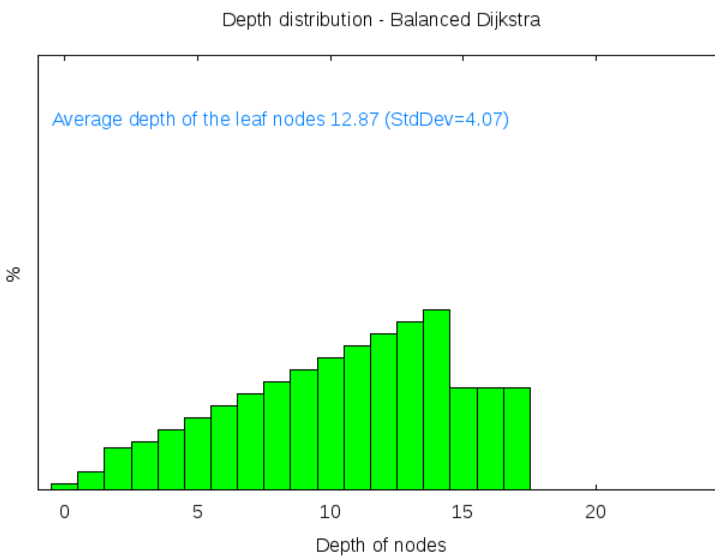

Figure 18: Node depth distribution with Dijkstra and load balancing

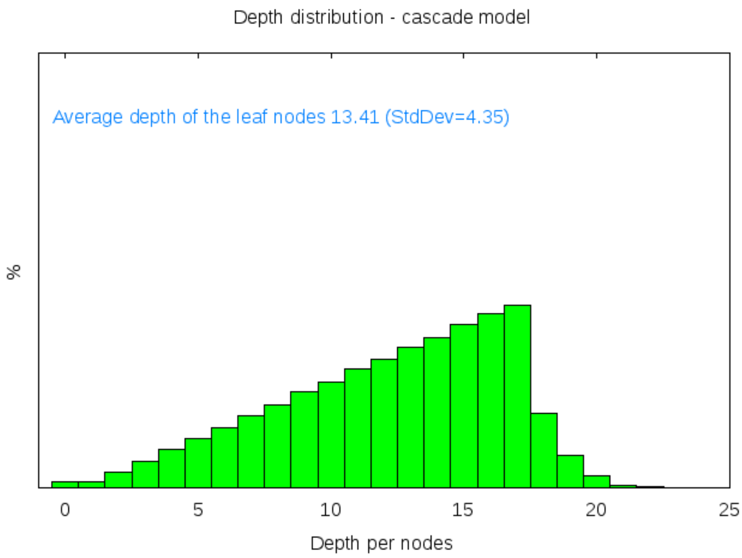

Figure 20: Node depth distribution with a cascade propagation model

and from the point of view of the static nature of our network nodes.

We have chosen a tree topology as the best choice for our deployment. Although, we found a limitation in the addresses assignment mechanism implemented for ZigBee. The latest versions of the ZigBee specification seem to have solved this limitation but we have proposed a solution based on the use of gateways. The gateways allow to apply hierarchical routing, add new nodes to the tree structure, and cover a large vineyard surface. We have implemented this solution in our simulations and we have obtained samples of different structures.

Finally, we have proposed a mechanism for deploying sensor networks based on the pre-assignment of addresses in order to optimize the network start and its structure. We have applied different models to generate these structures. Once we compared the results, we decided that the optimal model is the based on applying the modified Dijkstra algorithm for load balancing of nodes. In any case it would be appropriate to compare this result by a study of energy consumption.

[1] C. R. Cunha, E. Peres, R. Morais, M. C. Reis, Enhancing tourism through viticulture enterprises in douro region: The inov@ douro model (2010).

[2] N. Bulusu, D. Estrin, L. Girod, J. Heidemann, Scalable coordination for wireless sensor networks: self-configuring localization systems, in: International Symposium on Communication Theory and Applications (ISCTA 2001), Ambleside, UK.

[3] E. Shih, S.-H. Cho, N. Ickes, R. Min, A. Sinha, A. Wang, A. Chandrakasan, Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks, in: Proceedings of the 7th Annual International Conference on Mobile Computing and Networking, MobiCom '01, ACM, New York, NY, USA, 2001, pp. 272–287.

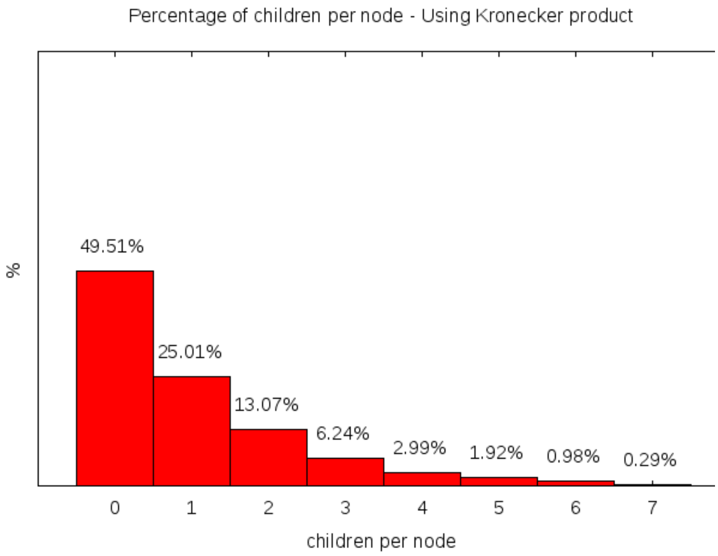

Figure 21: Distribution of children per node using the Kronecker product with matrix 1

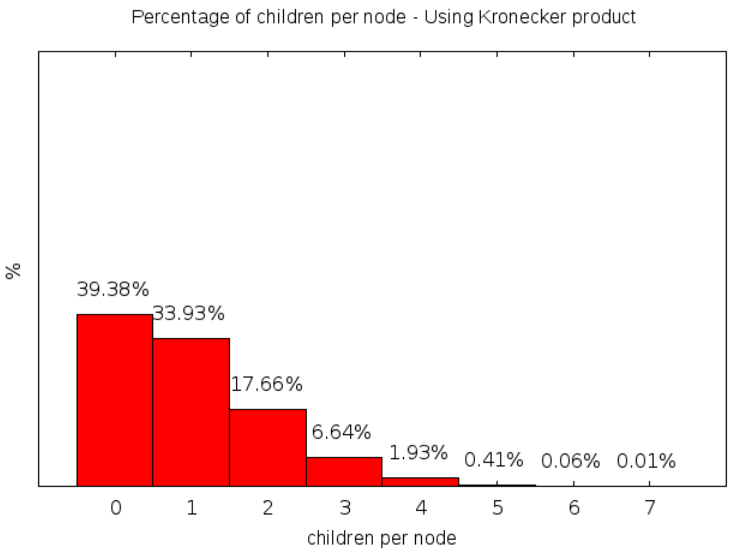

Figure 23: Distribution of children per node using the Kronecker product with matrix 2

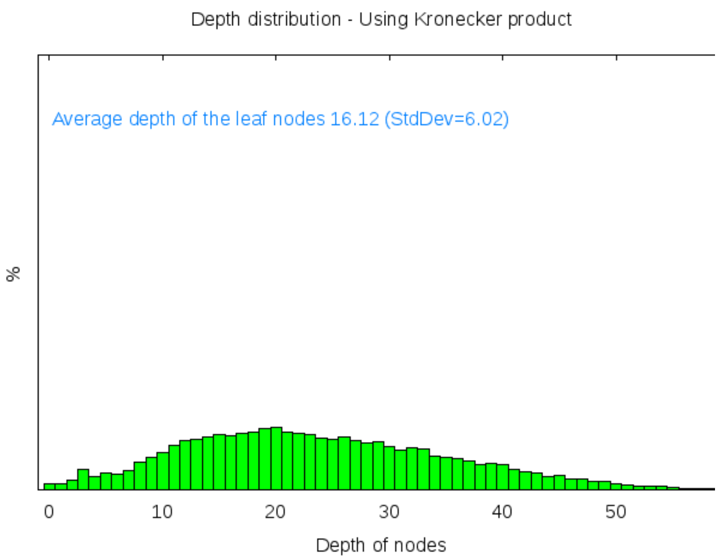

Figure 22: Node depth distribution using the Kronecker product with matrix 1

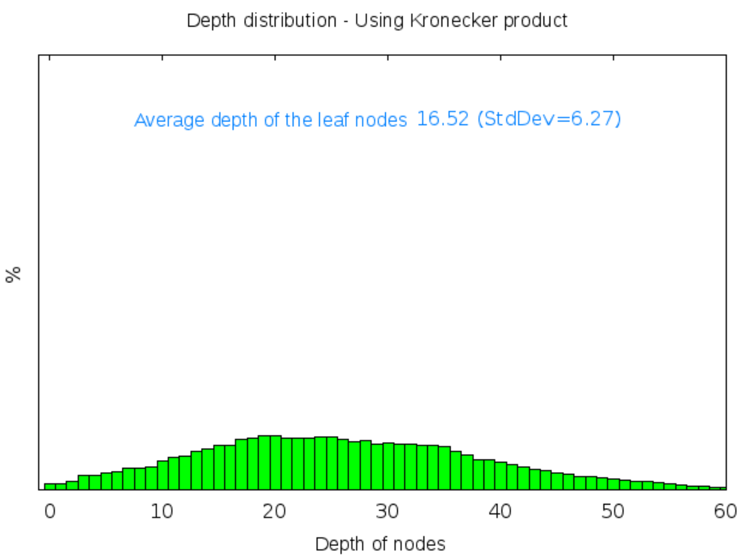

Figure 24: Node depth distribution using the Kronecker product with matrix 2

[4] M. A. Labrador, P. M. Wightman, Topology Control in Wireless Sensor Networks: with a companion simulation tool for teaching and research, Springer Science & Business Media, 2009.

[5] G. Hoblos, M. Staroswiecki, A. Aitouche, Optimal design of fault tolerant sensor networks, in: Control Applications, 2000. Proceedings of the 2000 IEEE International Conference on, pp. 467–472.

[6] W. K. G. Seah, Z. A. Eu, H. Tan, Wireless sensor networks powered by ambient energy harvesting (WSN-HEAP) - survey and challenges, in: Wireless Communication, Vehicular Technology, Information Theory and Aerospace Electronic Systems Technology, 2009. Wireless VITAE 2009. 1st International Conference on, pp. 1–5.

[7] R. Morais, M. A. Fernandes, S. G. Matos, C. Serôdio, P. J. S. G. Ferreira, M. J. C. S. Reis, A ZigBee multi-powered wireless acquisition device for remote sensing applications in precision viticulture, Computers and Electronics in Agriculture 62 (2008) 94–106.

[8] Y. Xu, J. Heidemann, D. Estrin, Geography-informed energy conservation for ad hoc routing, in: ACM MOBICOM, pp. 70–84.

[9] J. N. Al-Karaki, A. E. Kamal, Routing techniques in wireless sensor networks: a survey, Wireless Communications, IEEE 11 (2004) 6–28.

[10] R. Rajagopalan, P. K. Varshney, Data aggregation techniques in sensor networks: A survey (2006).

[11] J. Robinson, J. Harding, The Oxford Companion to Wine, Oxford University Press, 2006.

[12] A. M. Ortiz, T. Olivares, L. Orozco-Barbosa, M. Perez-Juana, Measurements with different role-based wireless sensor network organizations, in: Proceedings of the 3Nd ACM Workshop on Performance Monitoring and Measurement of Heterogeneous Wireless and Wired Networks, PM2HW2N '08, ACM, New York, NY, USA, 2008, pp. 9–16.

[13] Y. Zhang, N. Meratnia, P. Havinga, Outlier detection techniques for wireless sensor networks: A survey, Communications Surveys Tutorials, IEEE 12 (2010) 159–170.

[14] L. Buttyán, P. Schaffer, I. Vajda, RANBAR: RANSAC-based resilient aggregation in sensor networks, in: Proceedings of the Fourth ACM Workshop on Security of Ad Hoc and Sensor Networks, SASN '06, ACM, New

York, NY, USA, 2006, pp. 83–90.

[15] Taylor, C. Acevedo-Opazo, H. Ojeda, B. Tisseyre, Identification and significance of sources of spatial variation in grapevine water status, Australian Journal of Grape and Wine Research 16 (2010) 218–226.

[16] F. J. Riquelme, A. B. Ramos, F. M. Garc'ia, L. O. Barbosa, J. Garc'ia-Consuegra, F. Q. Flor, Determinación de la variabilidad vitivinícola en parcela mediante redes inalámbricas de sensores y su aplicación en viticultura de precisión: El proyecto wisevine (2005-2007) (????).

[17] C. A. Cambardella, D. L. Karlen, Spatial analysis of soil fertility parameters, Precision Agriculture 1 (1999) 5–14.

[18] A. P. Mallarino, E. S. Oyarzabal, P. N. Hinz, Interpreting Within-Field relationships between crop yields and soil and plant variables using factor analysis, Precision Agriculture 1 (1999) 15–25.

[19] A. Ortiz, T. Olivares, L. Orozco-Barbosa, A heterogeneous role-based sensor network, in: Proceedings of the 2Nd ACM Workshop on Performance Monitoring and Measurement of Heterogeneous Wireless and Wired Networks, PM2HW2N '07, ACM, New York, NY, USA, 2007, pp. 61–67.

[20] C. Jardak, K. Rerkrai, A. Kovacevic, J. Riihijarvi, P. Mahonen, Design of large-scale agricultural wireless sensor networks: email from the vineyard, International Journal of Sensor Networks 8 (2010) 77–88.

[21] P. Ran, M. heng Sun, Y. min Zou, ZigBee routing selection strategy based on data services and Energy-Balanced ZigBee routing, in: Services Computing, 2006. APSCC '06. IEEE Asia-Pacific Conference on, pp. 400–404.

[22] F. Cuomo, S. Della Luna, U. Monaco, F. Melodia, Routing in ZigBee: Benefits from exploiting the IEEE 802.15.4 association tree, in: Communications, 2007. ICC '07. IEEE International Conference on, pp. 3271–3276.

[23] M.-S. Pan, Y.-C. Tseng, Quick convergecast in ZigBee beacon-enabled tree-based wireless sensor networks, Computer Communications 31 (2008) 999–1011. Mobility Management and Wireless Access.

[24] P. Baronti, P. Pillai, V. W. C. Chook, S. Chessa, A. Gotta, Y. F. Hu, Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards, Computer Communications 30 (2007) 1655–1695. Wired/Wireless Internet Communications.

[25] Y. rui Li, H. bin Shi, B. yu Tang, Address assignment and routing protocol for Large-Scale uneven wireless sensor networks, in: Computer Network and Multimedia Technology, 2009. CNMT 2009. International Symposium on, pp. 1–4.

[26] M. Fang, J. Wan, X. Xu, A preemptive distributed address assignment mechanism for wireless sensor networks, in: Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on, pp. 1–5.

[27] H.-J. Ju, D. Ko, S. An, Enhanced distributed address assignment scheme for ZigBee-based wireless sensor networks, in: Intelligent Networking and Collaborative Systems, 2009. INCOS '09. International Conference on, pp. 237–240.

[28] G. Bhatti, G. Yue, A structured addressing scheme for wireless multihop networks, Technical Report, Technical Report, Mitsubishi Electric Research Labs Cambridge, MA, 2005.

[29] A. D. Siuli Roy, S. Bandyopadhyay, Agro-sense: Precision agriculture using sensor-based wireless mesh networks, in: Innovations in NGN: Future Network and Services, 2008. K-INGN 2008. First ITU-T Kaleidoscope Academic Conference, pp. 383–388.

[30] G. Yang, Z. Wang, W. Wang, X. He, An improved ETR protocol with energy awareness for wireless sensor networks, in: Mobile Ad-hoc and Sensor Networks, 2009. MSN '09. 5th International Conference on, pp. 117–123.

[31] Sharad, S. Mishra, A. Sharma, D. S. Chauhan, A different approach of addressing, energy efficient routing and data aggregation for enhanced tree routing protocol 306 (2012) 306–321.

[32] V. Jacobson, M. Mosko, D. Smetters, J. Garcia-Luna-Aceves, Content-centric networking, Whitepaper, Palo Alto Research Center (2007) 2–4.

[33] A. Varga, R. Hornig, An overview of the OMNeT++ simulation environment, in: Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops, Simutools '08, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, Belgium, 2008.

[34] E. Sadikov, M. Medina, J. Leskovec, H. Garcia-Molina, Correcting for missing data in information cascades, in: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, WSDM '11, ACM, New York, NY, USA, 2011, pp. 55–64.

[35] J. Leskovec, Stanford network analysis platform (snap), 2006.

[36] J. Leskovec, D. Chakrabarti, J. Kleinberg, C. Faloutsos, Z. Ghahramani, Kronecker graphs: An approach to modeling networks, J. Mach. Learn. Res. 11 (2010) 985–1042.