

Lifetime and Reliability Evaluation Models based on the Nearest Closer Protocol in Wireless Sensor Networks

Ning Cao Russell Higgs Gregory M.P. O'Hare Rui Wu
CLARITY: The Centre for Sensor Web Technologies
University College Dublin, Belfield, Dublin 4, Ireland

{ning.cao, russell.higgs, gregory.ohare}@ucd.ie, rui.wu.1@ucdconnect.ie

Abstract—This paper aims to introduce some key parameters for the tracking application in wireless sensor networks. In this paper, the Nearest Closer protocol has been implemented in J-sim simulation platform, and consequently some useful trade-off analysis results among the density, reliability and lifetime have been obtained. Based on these results, two evaluation models are proposed.

Index Terms—J-Sim; Nearest Closer; Evaluation Model

I INTRODUCTION

A WIRELESS sensor network consists of a number of sensors deployed either randomly or in a pre-determined state in a given two or three dimensional space, thus forming a network of sensors. The sensors are designed to measure one or more physical quantities in the space, such as temperature or location. The sensors need to transmit this collected data to the end-user, who often will be outside of the space being measured, which could well be a dangerous environment. As the sensors concerned are wireless they are typically powered by a battery with a finite lifetime and power output, it may be impossible or impracticable to recharge or replace such batteries. Thus in a real wireless sensor network a number of parameters naturally need to be considered such as energy consumption, network lifetime and network throughput. The network may be assigned a routing protocol. There exist a lot of routing protocols, so to choose the suitable routing protocol has become a significant problem.

The routing protocols can be divided into three main types: flat protocols, location-based protocols and hierarchical protocols. Single-hop, Nearest Closer and LEACH are typical and basic routing protocols for these three types respectively.

We have integrated the Single-hop and LEACH [1] protocols into the simulation tool J-Sim and provided mathematical models for each protocol. This paper will focus on the multi-hop routing protocol.

The multi-hop protocol we realized in J-Sim is called Nearest Closer protocol. To implement this protocol each node has to know: its own position, the position of its neighbours within its transmission range and the position of the sink node. The main idea in the Nearest Closer protocol is that the distance between the receiver sensor node and the sink node is shorter than the distance between the transmitter sensor and

the sink node. In addition, the transmitter sensor will transmit to its nearest neighbour that is closer to the sink node.

In this paper, the relationships among density, lifetime, and reliability will be investigated for the Nearest Closer protocol by simulating the tracking application. Based on the results of these parameters, two intelligent evaluation models will be proposed. This means that wireless sensor network users can predict lifetime and reliability directly. Thus sensor nodes can be deployed in such a network without further simulations.

The rest of this paper is structured as follows: Section 2 describes some details of J-Sim. Section 3 defines the evaluation parameters. Section 4 focuses on the experimental set-up information. Section 5 describes the results which have been obtained so far. Section 6 proposes the Lifetime model. Section 7 proposes the Reliability model. Section 8 concludes this paper.

II J-SIM

J-Sim [2] (formerly known as JavaSim) is an open-source, component-based compositional network simulation environment. The system is based on the IEEE 802.11 [3-4] implementation provided with J-Sim. IEEE 802.11 is the first wireless LAN (WLAN) standard proposed in 1997. J-Sim is implemented on top of the autonomous component architecture (ACA), components are the basic elements in this architecture and through these J-Sim implements the data transmission process. J-Sim provides a script interface that allows its integration with Tcl, and has been developed entirely in the Java platform. Java is a general purpose object-oriented computing language that is specifically designed to have as few implementation dependencies as possible.

This work selected J-Sim as its simulation tool for the following reasons:

The authors of J-Sim have performed detailed performance comparisons in simulating several typical WSN scenarios in J-Sim and ns-2. The simulation results indicate J-Sim and ns-2 incur comparable execution time, but the memory allocated to carry out simulation in J-Sim is at least two orders of magnitude lower than that in ns-2. As a result, while ns-2 often suffers from out-of-memory exceptions and was unable to carry out large-scale WSN simulations, the proposed WSN framework in J-Sim exhibits good scalability.

J-Sim models are easily reusable, so users can combine the components in the framework freely. J-Sim also provides a Graphical User Interface (GUI), which makes it easy to operate the simulation.

J-Sim is a Java-based platform. The Java-based sensors could be integrated with Java-based simulation tools in the future.

III. EVALUATION PARAMETERS

In this section, the evaluation parameters of Reliability Lifetime and Density will be defined for use in the following sections. This paper will focus on the parameters of Energy, Density, Lifetime and Reliability. Based on the results of experiments measuring these parameters, an intelligent evaluation model will eventually be constructed.

This work has completed power control over the radio components on the J-Sim simulation platform, which makes simulation of power consumption possible.

There is no need to assume that all sensors in a WSN measure the same phenomena; however, in this paper it is assumed that all sensors can transmit data about the phenomena they detect and can receive and retransmit data from other sensors about any of the phenomena detected in the system.

A. Reliability

In this work, experiments are conducted using the concept of reliability, defined by:

Reliability = the number of packets received by the sink node / the number of packets sent to the sink node

The advantages of using reliability are two-fold. Firstly, this definition can be used in the field and laboratory. In the field provided the sink node (end-user) knows how many cluster heads or sensors there are and the average number of packets of sensed data per sensor per time unit, then reliability can be estimated at any time during the lifetime of the network using the number of time units elapsed. Any difference between reliability measured in the field and laboratory could be used to detect how many packets of data are lost between sensors and cluster heads or the sink node (this would include counting sensors that have failed or have otherwise been lost to the system). Secondly, this simple definition of reliability can be used in a network where there are sensors of various types monitoring different things to produce a single simple measure of accuracy. In this situation our definition gives a systemic measure of accuracy for the network as a whole independent of the applications. This definition also makes it easy to analyze communication among the sensor nodes; in particular it makes the estimation of data collision in the wireless sensor network possible.

A frequently used definition for tracking accuracy is to set the tracking error equal to the Euclidean distance between the estimated and actual locations of the target.

The chief advantage of our definition for the tracking application, that this paper is going to simulate, is that it is independent of the target position.

B. Lifetime

Network lifetime has become the key characteristic for evaluating sensor networks in an application-specific way. In particular the availability of nodes, and connectivity have been included in discussions on network lifetime. In fact, even quality of service measures can be reduced to lifetime considerations. Network lifetime is the time span from the deployment of the sensors to the instant when the network is considered non-functional. When a network should be considered non-functional is application specific. It could be, for example, the instant when the first sensor dies, a percentage of sensors die. Conserving sensor energy and increasing tracking accuracy are the two main goals for the research of target tracking applications in wireless sensor networks. Simulation is a common way to compare these two parameters. Before the real deployment of sensor nodes, users always perform some simulation tests for the tracking application. In the simulations, finding the trade-off point between energy consumption and tracking accuracy is one of the key questions to be addressed. In addition, evaluation analysis will be performed here among all the related parameters.

Sensors need to send packets to other sensors or sink node in a WSN. On the other hand, the sink node will receive packets from sensors. Thus, the definition for network lifetime we have taken in our experiments is the time when the last packet is received by the sink node. Last packet here means the sink node cannot receive any packet from sensors after receiving this packet.

C. Density

The number of sensors deployed in a fixed area will be taken as the Density parameter in this paper. Obviously as the number of sensors increases, so does the average number of sensors per square metre.

IV. EXPERIMENTAL SET-UP

The simulated area for the experiments in the following sections is a 10 metre \times 10 metre square with randomly deployed nodes. The sink node for this application is located in the middle of this square. One of the primary reasons for selecting this setup is to allow the results to be generalized to large areas by concatenation of networks similar to this. For example, a 50 metre \times 50 metre square region could be configured using 25 instances of the setup used here in a 10 \times 10 grid formation. All the points (in the figures) in the following section are the average value from at least 5 separate experiments.

In the following section, reliability and lifetime will be evaluated using density as a parameter. As the number of nodes increases in a fixed space, the efficiency of the sensor network may become better or worse. So in some simulations the researchers may care about the number of sensors they are using in the network. It is obvious that

some of the nodes will be out of power in a real wireless sensor network. Thus attention will move to the number of sensors that are alive.

V. RESULTS

A series of experiments were carried out with the number of sensors starting at 10, and increasing in increments of 10 to 300 sensors. The transmission radius for each sensor was fixed at 15 metres.

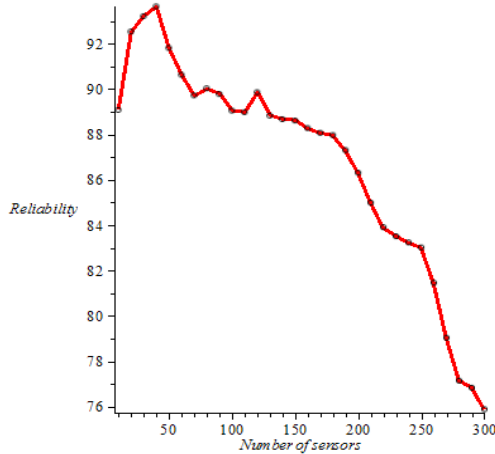


Figure 1. Density, Reliability relationship for NC.

In Fig. 1, the relationship between the number of sensors and Reliability is very clear. Reliability for this application increased as the number of sensors increased to 40, when it reached its highest value. It then essentially decreased as the number of sensors increased from 40 to 300, when it reached its lowest value. This may be explained by observing that as the number of nodes increases in the fixed space more sensors will join the data transmission process and consequently communication among the sensors will become more and more complex. Consequently dropped data due to data collision and latency cannot be ignored. We conclude that it is reasonable to expect reliability to decrease with density. In addition, the variance for Reliability is 23.3978 and the standard deviation for Reliability is 4.83713.

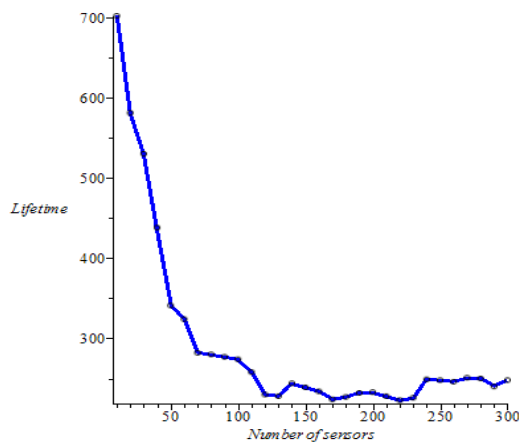


Figure 2. Density, Lifetime relationship for NC.

In Fig. 2, Lifetime reached its lowest value when the number of sensors equalled 220, whereas the highest value of Lifetime occurred when there were 10 sensors. When the number of sensors equalled 40, Reliability was 93.64%, its highest level, but with this number Lifetime is fairly short. This illustrates that it is possible for users to choose an optimum density value for this application depending on the Reliability and Lifetime required. In addition, the variance for Lifetime is 13602.64483 and the standard deviation for Lifetime is 116.63038.

VI. LIFETIME MODEL

Hop count is one way to measure the energy requirement of a routing task, thus using a constant metric per hop. However, if nodes can adjust their transmission power (knowing the location of their neighbours) the constant metric can be replaced by a power metric $u(d) = e + d^\alpha$ (or some variation of this) for some constants α and e that depend on the distance d between nodes. The value of e , which includes energy loss due to start up, collisions, retransmissions, and acknowledgements, is relatively significant, and protocols using any kind of periodic hello messages are extremely energy inefficient.

The basic idea for the NC protocol model to be constructed in this section is that the energy consumed by an average sensor in sending all its data to its nearest neighbour is of the form $w(s)E(d^2)$, where $E(d^2)$ is the expected value of the square of the distance between sensors and $w(s)$ is the average number (weight) of packets of data to be sent to its nearest neighbour. This figure also represents the average amount of energy to send one packet of data to the sink node.

Now in the NC protocol one possible parameter that could affect Lifetime is the number of paths or trees formed by the sensors. This is not the case as demonstrated by the following experiments, which are of independent interest. Using J-Sim it is difficult to extract the paths that sensors transmit along to the sink node. For this reason a program was written in the algebraic software package Magma [5] to not only compute the individual paths, but to marry them together to form the initial distinct trees.

Of course during the Lifetime of the network various sensors will die, normally from the centre outwards, and new trees will be formed. The mean number of trees corresponding to different numbers of sensors is given in Table 1 below. This mean number of trees was found by running the program 1,000 times for each given number of sensors. It could be seen from this table that the number of trees is almost independent of the number of sensors (excluding very small values) and is of the order of 2.55 trees.

TABLE I
MEAN NUMBER OF TREES

No. of sensors	Trees
10	2.537
20	2.527
30	2.561
40	2.559
50	2.531
60	2.550
70	2.542
80	2.534
90	2.553
100	2.549
110	2.568
120	2.578
130	2.555
140	2.543
150	2.574
160	2.550
170	2.557
180	2.552
190	2.575
200	2.582
210	2.574
220	2.600
230	2.571
240	2.573
250	2.569
260	2.551
270	2.575
280	2.543
290	2.563
300	2.562

The conclusion from Table 1 is that the mean number of trees is independent of the number of sensors and therefore can be discounted as having any effect on Reliability or Lifetime. The reason that on average two or three trees occur can be satisfactorily explained by considering the diagram below.

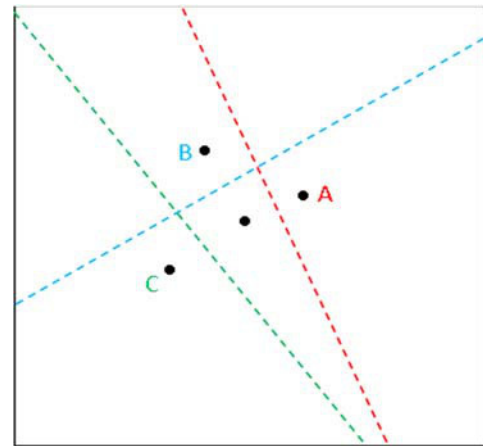


Figure 3. Typical Tree Zones

In Fig. 3 the points A, B and C represent the three nearest sensors to the sink node in the centre of the square amongst the randomly distributed sensors in the region. The perpendicular bisectors of the line segments joining A, B and C to the sink node have then been constructed. The next furthest sensor will with high probability be in one of the three zones containing A, B or C and its corresponding perpendicular bisector. Thus, in this diagram that three trees will probably result with A, B and C as the roots. Then, in 'zone' A it may be the case that further out in the tree that sensors cross over into zone B or C. It is fairly easy to draw diagrams where two or four trees will result, but only a very lop-sided distribution will result in just one tree if a reasonable number of sensors are being distributed.

For the NC protocol the relation 'is the closest neighbour to' is not symmetric, that is B can be the closest neighbour to A without A being the closest neighbour to B. Thus for this protocol it is necessary to find the closest neighbour to a sensor A, but with the stipulation that the neighbour is closer to the sink node than A. It is also the case that the sink node could be the closest neighbour to A, but the sink node is a fixed point and therefore does not fit in with the assumption that this work is dealing with a random distribution of sensors.

The expected distance d between a sensor node and its closest neighbour that is closer to the sink node (including the sink node itself) is given in the second column of the table and the expected distance of the square of the distance is given in the third column. Finally in the fourth column the average weight of the sensors is given, where each branch in the various trees count as weight one. This is then a measure of the number of packets of data an average sensor has to transmit to its nearest neighbour and it is also the average number of hops from the sensors to the sink node. The results of these simulations are tabulated in Table 2.

TABLE II
EXPECTED DISTANCES AND WEIGHTS, VARIABLE SENSORS

Sensors	$E(d)$	$E(d^2)$	$w(s)$
10	2.1945	6.0533	2.3526
20	1.5985	3.2761	3.3421
30	1.3223	2.2444	4.0821
40	1.1512	1.7023	4.7152
50	1.0264	1.3535	5.2218
60	0.9402	1.1350	5.7941
70	0.8672	0.9679	6.2249
80	0.8104	0.8450	6.6531
90	0.7657	0.7545	7.0685
100	0.7256	0.6759	7.4110
110	0.6926	0.6155	7.7859
120	0.6618	0.5634	8.1589
130	0.6360	0.5200	8.4701
140	0.6124	0.4817	8.7956
150	0.5910	0.4485	9.0999
160	0.5724	0.4210	9.4502
170	0.5553	0.3957	9.7218
180	0.5394	0.3734	9.9967
190	0.5240	0.3525	10.3059
200	0.5108	0.3346	10.5684
210	0.4983	0.3183	10.7823
220	0.4874	0.3046	11.0709
230	0.4759	0.2904	11.2847
240	0.4658	0.2785	11.5108
250	0.4562	0.2670	11.8269
260	0.4466	0.2559	12.0056
270	0.4393	0.2475	12.2536
280	0.4308	0.2378	12.4573
290	0.4231	0.2296	12.7054
300	0.4160	0.2216	12.9399

It should be noted that for those sensors whose nearest neighbour is the sink node that the distribution of the distance and distance squared to the sink node is slightly different to those between sensors, and the averages are always slightly higher than those given in Table 2, with the exception of the 10 sensor case when there is a significant difference.

Two results emanate from this table. The first is that $w(s)E(d)$ is nearly constant as it only ranges from 5.16 to 5.45 and is a measure of the average distance from a sensor to the sink node measured along the hop path.

The second is that $w(s)E(d^2)$ decreases as the number of sensors increases, this shows that the energy expended by a sensor in receiving and sending received packets to its nearest neighbour decreases as the number of sensors increases. It also represents a measure of the average amount of energy needed to transmit one packet of data to the sink node. Two theories are proposed to explain why the lifetime decreases as the number of sensors increases.

A. Early Termination

To explain early termination it is useful to consider the situation when two trees occur and the nearest (root) sensors to the sink node. These two sensors have to receive and transmit all the sensor data to the sink node and therefore have long transmission times with one time slot per packet. A transmission break, which in our definition of lifetime is interpreted as the end of lifetime, could result when both of the root sensors do not possess sufficient energy to transmit their data to the sink node. In a sense this is like the situation for the LEACH protocol with the root sensors acting as fixed cluster heads and the lifetime being controlled by the lifetimes of those sensors. A transmission break could also result if there is a catastrophic data collision in which all the packets of data from the root sensors collide with each other at the sink node and from which the system cannot recover. It is highly probable in a large system that some data collisions will occur at the sink node.

B. Data build up

As explained above the tree root sensors have long transmission times to transmit their data in a number of slots. Each sensor can transmit and receive data, but cannot do both simultaneously, also in the slotted ALOHA [6-8] MAC design transmissions in a tree are sequenced. So the sensors immediately prior to a root sensor will receive more data than previously when the root sensor transmits, which will subsequently be transmitted to the root sensor and then the pattern will be repeated. The situation will become more complicated higher up the tree, but generally the amount of data to be transmitted by a sensor will increase with time in a non-linear manner. In particular the root sensors will have more and more packets of data to transmit as data builds up in the tree. So for a fixed number of sensors the amount of energy required for a root sensor to transmit its packets of data to the sink node will increase with time. Delays in the slotted ALOHA protocol will lead to data build up and data collisions will result in retransmissions.

If a varying number of sensors is considered then the amount of energy needed to send a frame of packets may increase with the number of sensors despite the fact that $w(s)E(d^2)$ decreases with the number of sensors. The energy required to transmit such a frame of data (in reality such a frame may not be transmitted in consecutive slots) will increase with the number of sensors if the frame length (the number of packets) exceeds the ratio between $E(d^2)$ for the corresponding number of sensors; this latter ratio is always less than 1.85 between successive values in Table 2 and approaches 1 as the number of sensors

increase, so that it is reasonable to expect that the corresponding ratio of frame lengths will exceed this. This effect is balanced by the fact that in a smaller system sensor will transmit and receive data more often than in a larger system. It is also of note that most energy is used in data transmission rather than reception, however, energy used for data reception could be significant when compared to energy use in a small system.

The conclusion is that $w(s)$ may not be the correct weight to be attached to an average sensor, but instead represents a lower bound for the weight.

The two theories can be related in that with data build up the root sensors may end up transmitting a large amount of data to the sink node using a lot of energy in the process. This could ultimately lead to root sensors having insufficient energy to transmit the frame or a catastrophic data collision, in either case this will lead to early termination of the system. In reality both theories proposed here contribute to the lifetime figures obtained with the latter theory difficult to quantify in a model because of the complexity of the trees.

Anyway lifetime in general can be modelled by only considering the roots of the normally two or three trees. The case of just one tree occurring is of interest for a small number of sensors, for in this scenario the system should run until the only root sensor dies naturally, at that time the system might interpret the death as a transmission break or the next sensor(s) out will take over the role of the root(s). The basic model constructed here is of the form

$$\frac{E(T)K}{nE(d^2)}$$

where n is the number of sensors, K is the total initial energy of a sensor and $E(T)$ is the expected number of trees. Here $n/E(T)$ represents the average weight per tree and since we know that $E(T)$ is nearly constant, the model should roughly resemble the curve $1/n$. Our model in particular assumes that the sensors are evenly distributed amongst the trees and disregards data build up, it should therefore represent an upper bound for the lifetime once the system is running at full capacity. This would imply that lifetime decreases monotonically as the number of sensors increases; however, this basic model assumes that the system runs at full capacity from the start and ignores idle slots in the slotted ALOHA design. Thus there is a base value below which lifetime will not fall, as the system will behave like a smaller system during its initial phase and there will also always be a number of idle slots, this base value will be taken to be the smallest lifetime value of 223 that occurs when $n = 220$. So the revised model is

$$(1 - \lambda(n))223 + \lambda(n)\frac{E(T)K}{nE(d^2)}$$

where $\lambda(n)$ is a proportionality measure with $\lambda(220) = 0$. Now from the lifetime data $\lambda(n)$ will be very close to 0 for all $n \geq 120$. Setting $\lambda(10) = 1$

yields the value $K = 16750$. Computation gives the following values for $\lambda(n)$:

TABLE III
EVALUATION OF THE PROPORTIONALITY MEASURE

n	10	20	30	40	50
$\lambda(n)$	1	0.846	0.741	0.529	0.292
n	60	70	80	90	100
$\lambda(n)$	0.250	0.146	0.141	0.133	0.125
n	110	120	130	140	150
$\lambda(n)$	0.085	0.017	0.012	0.051	0.038
n	160	170	180	190	200
$\lambda(n)$	0.027	0.002	0.009	0.021	0.024
n	210	220	230	240	250
$\lambda(n)$	0.012	0	0.007	0.062	0.059
n	260	270	280	290	300
$\lambda(n)$	0.055	0.064	0.065	0.040	0.059

The parameter $\lambda(x)$ can be piecewise approximated by

$$1.1438571428571 - 0.01501071428514x$$

for $10 \leq x \leq 70$, and

$$0.3954 - 0.0029514285714286x$$

for $80 \leq x \leq 130$ by fitting straight lines through the given values. An alternative that will yield better local approximations is to interpolate between the known values of λ . Thus if $n \leq x < n + 10$, then

$$\lambda(x) \cong \left(1 - \frac{x - n}{10}\right)\lambda(n) + \frac{x - n}{10}\lambda(n + 10).$$

The general equation obtained for $10 \leq x \leq 133$ is

$$(1 - \lambda(x))223 + \lambda(x)\frac{42712.5}{xE(d^2)}$$

and the value of $E(d^2)$ for x sensors can again either be interpolated between the known values in Table 2 or by using the fitted formula

$$\frac{4918141600.24}{6653x^2 + 68674970x + 125057800}$$

for $10 \leq x \leq 300$. Using the latter formula (which is a good fit) yields at worst an implicit equation for the lifetime for $10 \leq x \leq 130$.

Finally we are stating that lifetime is essentially constant for $x \geq 140$ with a value between 223 and 250 depending on the topology of the sensors.

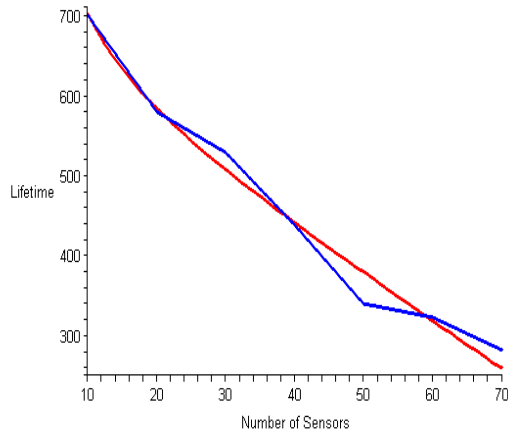


Figure 4 (a). Lifetime model (red), Lifetime (blue).

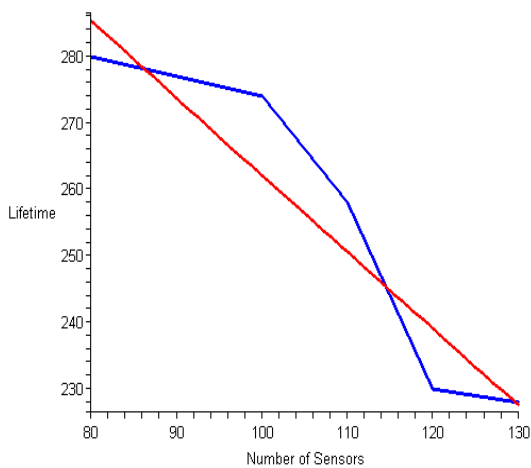


Figure 4 (b). Lifetime model (red), Lifetime (blue).

In both Fig 4 (a) and (b) the lifetime model is nearly a straight line and is clearly a better fit for ≤ 70 sensors.

VII. RELIABILITY MODEL

In the following Table 1 Reliability (as a percentage) divided by Lifetime is listed for the 15 metre transmission radius case, which gives (100 times) Reliability per unit Lifetime $R(n)/L(n)$:

TABLE IV
EVALUATION OF RELIABILITY PER UNIT LIFETIME

n	10	20	30	40	50
$R(n)/L(n)$	0.1269	0.1593	0.1759	0.2138	0.2692
n	60	70	80	90	100
$R(n)/L(n)$	0.2797	0.3182	0.3215	0.3241	0.3250
n	110	120	130	140	150
$R(n)/L(n)$	0.3449	0.3911	0.3896	0.3634	0.3708
n	160	170	180	190	200
$R(n)/L(n)$	0.3773	0.3932	0.3875	0.3763	0.3704
n	210	220	230	240	250
$R(n)/L(n)$	0.3728	0.3763	0.3695	0.3343	0.3348
n	260	270	280	290	300
$R(n)/L(n)$	0.3311	0.3149	0.3086	0.3202	0.3060

In the first of these regions the function $R(n)/L(n)$ may be approximated by

$$0.12609 + 0.002226n$$

or equivalently since we have determined $L(n)$ in the previous section

$$R(n) = 0.12609 * L(n) + 0.002226 * n * L(n)$$

for $0 \leq n \leq 120$.

This indicates that 12.5% of the Lifetime figure represents a lower bound for Reliability, but that Reliability per unit Lifetime will increase by about 2% of the Lifetime as the number of sensors is increased by 10.

In the next region the rate $R(n)/L(n)$ reaches its highest level and stays constant at about 0.3779, so that Reliability is about 38% of the Lifetime figure for $130 \leq n \leq 210$.

In the last region the rate decreases, this can be satisfactorily explained by the fact that Lifetime may be considered to be constant in this region and thus increasing the number of sensors just increases the complexity of communication to the sink node and the rate can be approximated by

$$0.549282 - 0.000832666n$$

for $220 \leq n \leq 300$. So in this region 55% of the Lifetime is an upper bound for Reliability, but this decreases by about 0.8% for each extra 10 sensors.

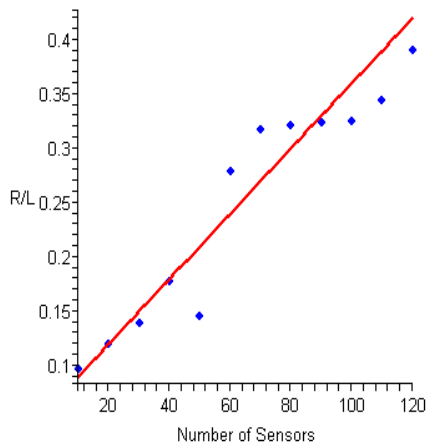


Figure 5 (a). R/L model (red), R/L data points (blue).

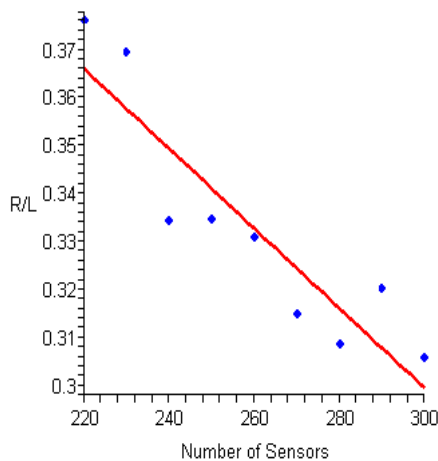


Figure 5 (b). R/L model (red), R/L data points (blue)

The model for R/L in Fig. 5 compared to the actual data point is fairly good with one exception of the point with 50 sensors, which either is genuinely exceptional or it may represent a data error.

VIII. CONCLUSIONS

In this paper, from a series of experiments using the J-Sim simulation tool several conclusions for NC were reached: (i) Reliability for this application increased as the

number of sensors increased to 40, when it reached its highest value. It then essentially decreased as the number of sensors increased from 40 to 300. (ii) Lifetime decreased as the number of sensors increased to 130 and then it fluctuated slightly as the number of sensors increased to 300. Lifetime reached its lowest value when the number of sensors equalled 220, whereas the highest value of Lifetime occurred when there were 10 sensors.

Based on these simulation results, two evaluation models among the parameters of lifetime, reliability, and density have been proposed. Thus, wireless sensor network users can predict the lifetime and reliability directly. This means that sensor nodes can be deployed in such a network without further simulations.

ACKNOWLEDGMENT

This Research is supported by Science Foundation Ireland under grant 07/CE/1147.

REFERENCES

- [1] N. Cao, R. Higgs and G.M.P. O'Hare. "An Intelligent Evaluation Model Based on the LEACH Protocol in Wireless Sensor Networks", *CyberC 2012*, pp. 381-388, Sanya, China, 2012. <http://dx.doi.org/10.1109/cyberc.2012.70>
- [2] A. Sobeih, W. Chen, J.C. Hou, L. Kung, N. Li, H. Lim, H. Tyan, and H. Zhang. "J-Sim: a simulation environment for wireless sensor networks", *Proceedings of the 38th Annual Symposium on Simulation*, pp. 175-187, IEEE Computer Society, Washington DC, 2005. <http://dx.doi.org/10.1109/anss.2005.27>
- [3] G. Bianchi. "Performance analysis of the IEEE 802.11 distributed coordination function", *IEEE Journal on Selected Areas in Communications*, 18 (3), pp. 535-547, 2000. <http://dx.doi.org/10.1109/49.840210>
- [4] S. Garg, M. Kappes. "An experimental study of throughput for UDP and VoIP traffic in IEEE 802.11b networks", *Wireless Communications and Networking*, 3, pp. 1748-1753, 2003. <http://dx.doi.org/10.1109/wcnc.2003.1200651>
- [5] W. Bosma, J. Cannon, and C. Playoust, "The Magma algebra system", *Journal of Symbolic Computation*, 24 (3), pp. 235-265, 1997. DOI: 10.1006/jsco.1996.0125
- [6] Y.C. Jenq. "On the stability of slotted ALOHA systems", *IEEE transactions on Communications*, 28 (11), pp. 1936-1939, 1980. <http://dx.doi.org/10.1109/tcom.1980.1094610>
- [7] A.B. Carleial, M.E. Hellman, "Bistable behavior of ALOHA type systems", *IEEE Transactions on Communications*, 23 (4), pp. 401-410, 1975. <http://dx.doi.org/10.1109/tcom.1975.1092823>
- [8] L.G. Roberts, "ALOHA packet system with and without slots and capture", *Computer Communication Review*, 5, pp. 28-42, ACM, 1975. <http://dx.doi.org/10.1145/1024916.1024920>